



אוניברסיטת בן-גוריון בנגב
הפקולטה למדעי ההנדסה
המחלקה להנדסת מכונות

בקרה ותכנון תנועה של קוואדרוטור

חיבור זה מהווה חלק מהדרישות
לקבלת התואר "מגיסטר" בהנדסה

מאת: שלמה שמש



אוניברסיטת בן-גוריון בנגב
הפקולטה למדעי ההנדסה
המחלקה להנדסת מכונות

בקרה ותכנון תנועה של קוואדרוטור

חיבור זה מהווה חלק מהדרישות
לקבלת התואר "מגיסטר" בהנדסה

מאת: שלמה שמש

מנחה: ד"ר אמיר שפירא

תאריך:

מחבר:

תאריך:

מנחה:

תאריך:

יו"ר ועדת מוסמכים:

תוכן עניינים

IV.....	תקציר	
VI.....	רשימת איורים	
VIII.....	רשימת גרפים וטבלאות	
VIII.....	רשימת גרפים	
VIII.....	רשימת טבלאות	
1.....	מבוא	1
3.....	סקר ספרות	2
5.....	פיתוח משוואות הדינמיקה של הכלי	3
5.....	3.1 המודל הדינמי של הקוואדרוטור	
6.....	3.2 המודל הקינטי של הקוואדרוטור	
6.....	3.2.1 דרגות החופש של הקוואדרוטור	
9.....	3.2.3 ביטויים למהירות הזוויתיות	
10.....	3.3 מציאת משוואות הדינמיקה באמצעות שיטת אוילר-לגראנז'	
12.....	3.4 הנחות ומעבר למודל פשוט	
14.....	4 בקרת עקיבה	4
15.....	4.1 מצב עקיבת אוריינטציה	
15.....	4.1.1 הגדרת בקר עבור מצב עקיבת אוריינטציה	
18.....	4.1.2 הוכחת יציבות עבור מצב עקיבת אוריינטציה	
23.....	4.2 מצב עקיבת מיקום	
23.....	4.2.1 הגדרת בקר עבור מצב עקיבת מיקום	
26.....	4.2.2 הוכחת יציבות עבור מצב עקיבת מיקום	
31.....	4.3 מצב בקרת מהירות	
32.....	4.4 סימולציות עבור מערכת הבקרה	
32.....	4.4.1 סימולציות עקיבה	
34.....	4.4.2 סימולציה עבור הגעה ממצב נתון למצב רצוי	
37.....	4.4.3 סיכום תוצאות הסימולציות	
38.....	5 תכנון מסלול	5
38.....	5.1 מרחב הקונפיגורציה	
38.....	5.1.1 הגדרת מרחב הקונפיגורציה	
39.....	5.1.2 מכשולים במרחב הקונפיגורציה	
40.....	5.1.3 מטרות תכנון המסלול במרחב הקונפיגורציה	
40.....	5.2 מפות דרכים	
41.....	5.2.1 מפת דרכים - הגדרה	

II

41	דפורמציה נסוגה : גרף וורונוי הכללי	5.3
41	5.3.1 דיאגרמת וורונוי	
42	5.3.2 גרף וורונוי הכללי	
44	5.3.3 דפורמציה נסוגה	
46	5.3.4 אלגוריתם ה - Brushfire	
49	5.4 האלגוריתם של דייקסטרא	
49	5.4.1 פעולת האלגוריתם	
51	5.4.2 סיבוכיות האלגוריתם	
52	5.5 סימולציות לתכנון מסלול	
52	5.5.1 סימולציות לבניית מרחב הקונפיגורציה	
55	5.5.2 סימולציות לבניית גרף וורונוי הכללי	
56	5.5.3 סימולציות למציאת המסלול הקצר ביותר באמצעות האלגוריתם של דייקסטרא	
58	5.5.4 סיכום תוצאות הסימולציות	
60	6 תכנון תנועה	
60	6.1 תכנון במרחב המצב	
60	6.1.1 הגדרת מרחב המצב	
61	6.1.2 אילוצים דיפרנציאליים	
62	6.1.3 מכשולים במרחב המצב	
63	6.1.4 מטרות תכנון התנועה במרחב המצב	
64	6.2 אלגוריתם ה - RRT	
64	6.2.1 פעולת האלגוריתם	
68	6.2.2 הוכחת שלמות הסתברותית	
69	6.2.3 הערכה לכמות האיטרציות הדרושה	
72	6.3 אלגוריתם ה - RRTT	
73	6.3.1 פעולת האלגוריתם	
76	6.3.2 הוכחת שלמות הסתברותית	
76	6.3.3 הערכה לכמות האיטרציות הדרושה	
77	6.4 סימולציות לתכנון תנועה	
77	6.4.1 סימולציות לתכנון תנועה באמצעות אלגוריתם ה - RRT	
80	6.4.2 סימולציות לתכנון תנועה באמצעות אלגוריתם ה - RRTT	
83	6.4.3 סיכום תוצאות הסימולציות	
86	7 סיכום ומסקנות	
88	8 ביבליוגרפיה	
91	נספח א' - ניסויים	
91	1.א בניית כלי הטיס	
91	1.1.א בניית גוף הכלי	
92	1.2.א הרכיבים האלקטרוניים והרכבתם	

III

95	א.3.1 ביצועי הדגם
98	א.2 זיהוי פרמטרים עבור כלי הטיס
98	א.2.1 מציאת קבועי העילוי והגרר
102	א.2.2 מציאת מומנטי האינרציה של כלי הטיס
107	א.3 מערכת הניסוי
107	א.4 תוצאות הניסויים

תקציר

בעבודה זו נציג אנליזה ומידול לדינמיקה של רחפן בעל ארבעה מדחפים (קוואדרוטור) תוך פיתוח משוואות הדינמיקה המלאות. בנוסף, נציג בקר לא-לינארי המבוסס על שיטות בגיאומטריה דיפרנציאלית, המאפשר עקיבה. לאחר מכן נציג שיטה קלאסית לתכנון מסלול המבוססת על מפת דרכים מסוג גרף וורונוי הכללי (Generalized Voronoi Graph) במרחב הקונפיגורציה, תוך שימוש באלגוריתם של דייקסטרא (Dijkstra's Algorithm). בנוסף, נציג את שיטת ה- RRT לתכנון תנועה במרחב המצב. בעבודה זו פותח אלגוריתם משופר לתכנון תנועה במרחב המצב, אלגוריתם ה- RRTT. בסופו של דבר נבצע סימולציות ונתאר את מערכת הניסוי שהוקמה.

הקוואדרוטור הוא כלי טיס בלתי מאויש (כטב"ם) המורכב מצלב מישורי ועליו ארבעה מנועים המניעים ארבעה רוטורים. תחילה, בוצע מידול לדינמיקה של הכלי ופותחו משוואות הדינמיקה המלאות של הכלי תוך התחשבות באפקטים הג'ירוסקופים, בזוויות הסיבוב של הרוטורים ובמודל המנועים. פיתוח משוואות התנועה נעשה באמצעות שיטת אויילר-לגראנז'. לאחר מכן בוצעו הנחות והזנחות על מנת להגיע למודל פשוט יותר איתו נוח לעבוד.

לקוואדרוטור ישנן למעשה ארבע כניסות אשר בעזרתן ניתן לשלוט על התנועה והאוריינטציה במרחב, כלומר ישנן ארבע כניסות עבור שש דרגות חופש. מערכת מסוג זה (יותר דרגות חופש מכניסות) נקראת מערכת תת-ממנועת (Underactuated) וקשה יותר לבקר מערכת מעין זו, וזאת משום שקיים צימוד בין היציאות. בעבודה זו נציג בקר לא-לינארי המבוסס על שיטות בגיאומטריה דיפרנציאלית הכולל שלושה מצבי טיסה שונים. כל מצב טיסה משיג עקיבה אקספוננציאלית אחרי ארבע דרגות חופש שונות - הוכחנו כי הוא משיג עקיבה כזו. בנוסף, בוצעו סימולציות להמחשת אופן פעולת הבקר.

לאחר שנציג את חוק הבקרה, נציג שיטה קלאסית לתכנון מסלול עבור מרחב עבודה מוכר. נציג שיטה לתכנון מסלול המבוססת על מפת דרכים מסוג גרף וורונוי הכללי במרחב הקונפיגורציה. בנוסף, נציג אלגוריתם חיפוש בגרף למציאת המסלול הקצר ביותר על גרף וורונוי הכללי - האלגוריתם של דייקסטרא. נראה הוכחה לכך כי גרף וורונוי הכללי הוא אכן מפת דרכים, נוכיח את שלמות האלגוריתם של דייקסטרא תוך חישוב הסיבוכיות שלו, ונראה את החסרונות שבתכנון מסלול (קונפיגורציה בלבד, ללא מהירויות) ביחס לתכנון תנועה (קונפיגורציה ומהירויות). כמו כן, בוצעו סימולציות לתכנון מסלול ע"פ שיטה זו.

בנוסף, נציג את שיטת ה- RRT לתכנון תנועה במרחב המצב תוך הסבר מפורט של אופן פעולת האלגוריתם, הוכחת שלמות הסתברותית והערכה לכמות האיטרציות הדרושה עד להגעה לתנועת פתרון. לאחר שנבין את היתרונות והחסרונות של אלגוריתם ה- RRT, נציג אלגוריתם חדש ומשופר - אלגוריתם ה- RRTT. נוכיח גם עבורו שלמות הסתברותית ונראה כי עבור מרחב עבודה סטנדרטי, נוכל להעריך כי כמות האיטרציות הדרושות להגעה לתנועת פתרון תהיה קטנה מזו הדרושה באלגוריתם ה- RRT. כמובן שבוצעו סימולציות עבור שתי השיטות השונות.

לסיום, נציג את תהליך הבנייה של הכלי. כמו כן, נציג ניסויים שנערכו בכדי לחשב את הפרמטרים עבור כלי הטיס. לסיכום, נציג את כלל מערכת הניסוי שהוקמה.

רשימת איורים

1	איור 1.1 - אוריינטציית הרוטורים ומבנה הקוואדרטור
5	איור 3.1 - דיאגרמת הכוחות והמומנטים הפועלים על הכלי
7	איור 3.2 - תיאור זוויות אוילר ואוריינטציית גוף הכלי במרחב
18	איור 4.1 - דיאגרמת הבלוקים עבור מצב בקרת אוריינטציה
25	איור 4.2 - דרך בחירת הווקטור \bar{x}_{b_c}
26	איור 4.3 - דיאגרמת הבלוקים עבור מצב בקרת מיקום
38	איור 5.1 - מרחב הקונפיגורציה עבור זרוע דו-פרקית
40	איור 5.2 - דוגמא לבניית מרחב קונפיגורציה ומציאת מסלול מתאים
42	איור 5.3 - דוגמא לדיאגרמת וורוני עבור אתרים רנדומליים
44	איור 5.4 - דוגמא לגרף וורוני הכללי במרחב דו-מימדי
45	איור 5.5 - דוגמא להומוטופיה בין שני מסלולים
47	איור 5.6 - דוגמא לבניית ה - GVG באמצעות אלגוריתם ה - Brushfire
48	איור 5.7 - ההגדרות השונות לסביבה קרובה לתא
48	איור 5.8 - שמירה על מרחק מנהטן ביחס לשימוש בשמונה תאים סמוכים
49	איור 5.9 - פסאודו-קוד לבניית ה - GVG באמצעות אלגוריתם ה - Brushfire
50	איור 5.10 - דוגמא למבנה של גרף בעל שבעה צמתים
50	איור 5.11 - המחשת מציאת המרחק הקצר ביותר באמצעות האלגוריתם של דייקסטרא
51	איור 5.12 - פסאודו-קוד למציאת המסלול הקצר ביותר באמצעות האלגוריתם של דייקסטרא
53	איור 5.13 - מרחב העבודה הדו-מימדי לו נבנה מרחב קונפיגורציה
53	איור 5.14 - מרחב העבודה התלת-מימדי לו נבנה מרחב קונפיגורציה
54	איור 5.15 - מרחב הקונפיגורציה שהתקבל עבור מרחב העבודה הדו-מימדי
54	איור 5.16 - מרחב הקונפיגורציה שהתקבל עבור מרחב העבודה התלת-מימדי
55	איור 5.17 - גרף וורוני הכללי שהתקבל עבור מרחב הקונפיגורציה הדו-מימדי
56	איור 5.18 - גרף וורוני הכללי שהתקבל עבור מרחב העבודה התלת-מימדי
57	איור 5.19 - הגרף המתקבל לאחר חיבור המוצא והיעד עבור מרחב הקונפיגורציה הדו-מימדי
57	איור 5.20 - המסלול המתקבל לאחר הפעלת האלגוריתם של דייקסטרא בדו-מימד
58	איור 5.21 - המסלול המתקבל לאחר הפעלת האלגוריתם של דייקסטרא בתלת-מימד
62	איור 6.1 - דוגמא להגדרת מצבים מסוכנים במרחב המצב
65	איור 6.2 - דוגמא לעץ רנדומלי נאיבי עם 2000 צמתים
65	איור 6.3 - דוגמא ל - RRT עם 2000 צמתים
66	איור 6.4 - פסאודו-קוד לתכנון תנועה באמצעות אלגוריתם ה - RRT
66	איור 6.5 - פסאודו-קוד עבור הפונקציה Extend
68	איור 6.6 - התקדמות המערכת בכיוון המצב x_{random}
69	איור 6.7 - דיאגרמת וורוני להמחשה של הוכחת השלמות ההסתברותית
71	איור 6.8 - המחשת אגן ואגני התכנסות
73	איור 6.9 - גידול שיח עגבניות באמצעות כלוב
74	איור 6.10 - אופן יצירת ה"כלוב" באמצעות גרף וורוני הכללי
75	איור 6.11 - פסאודו-קוד עבור אלגוריתם ה - RRT
75	איור 6.12 - פסאודו-קוד עבור הפונקציה Random_State2
77	איור 6.13 - מרחב העבודה עבור סימולציית תכנון התנועה הראשונה
78	איור 6.14 - אחד הפתרונות שהתקבלו באמצעות RRT עבור סימולציית התכנון הראשונה
79	איור 6.15 - אחד הפתרונות שהתקבלו באמצעות RRT עבור סימולציית התכנון השנייה
79	איור 6.16 - מרחב העבודה עבור סימולציית תכנון התנועה השלישית
80	איור 6.17 - אחד הפתרונות שהתקבלו באמצעות RRT עבור סימולציית התכנון השלישית
81	איור 6.18 - גרף וורוני הכללי והמסלול הקצר עבור סימולציית התכנון הראשונה
81	איור 6.19 - אחד הפתרונות שהתקבלו באמצעות RRT עבור סימולציית התכנון הראשונה

איור 6.20 - אחד הפתרונות שהתקבלו באמצעות RRTT עבור סימולציית התכנון השנייה	82
איור 6.21 - גרף וורנוי הכללי והמסלול הקצר עבור סימולציית התכנון השלישית	82
איור 6.22 - אחד הפתרונות שהתקבלו באמצעות RRTT עבור סימולציית התכנון השלישית	83
איור א.1 - פרופיל מוטות האלומיניום אשר שימשו לבניית שילדת הכלי	91
איור א.2 - תמונה של גוף הכלי הבנוי	92
איור א.3 - תמונה של מנוע Brushless Micro Motor	92
איור א.4 - תמונה של בקרי המהירות בהם השתמשנו	93
איור א.5 - תמונה של אופן חיבור בקר המהירות האלקטרוני למנוע	93
איור א.6 - החישן האולטרה-סוני אשר שימש כחישן גובה	93
איור א.7 - תמונה של המיקרו-בקרים המחוברים האחד לשני	94
איור א.8 - תמונה של פרופלור אשר אוזן באמצעות הדבקת פיסות דבק	94
איור א.9 - סכימת חיבור כלל הרכיבים האלקטרוניים	95
איור א.10 - תמונה של כלי הטיס הסופי שנבנה במהלך טיסה באוויר	95
איור א.11 - תמונה של מערכת הניסוי למדידת קבוע העילוי	98
איור א.12 - צורת חיבור המטוטלת לצורך מדידת מומנטי האינרציה של כלי הטיס	102
איור א.13 - תמונה הממחישה את אופן תליית הכלי כמטוטלת	103
איור א.14 - מבנה מערכת הניסוי	102

רשימת גרפים וטבלאות

רשימת גרפים

גרף 4.1 - מיקום מרכז המסה עבור סימולציית העקיבה הראשונה	32
גרף 4.2 - זוויות RPY עבור סימולציית העקיבה הראשונה	33
גרף 4.3 - מיקום מרכז המסה עבור סימולציית העקיבה השנייה	33
גרף 4.4 - זוויות RPY עבור סימולציית העקיבה השנייה	34
גרף 4.5 - מיקום מרכז המסה עבור הסימולציה הראשונה למעבר בין מצבים	35
גרף 4.6 - זוויות RPY עבור הסימולציה הראשונה למעבר בין מצבים	35
גרף 4.7 - מיקום מרכז המסה עבור הסימולציה השנייה למעבר בין מצבים	36
גרף 4.8 - זוויות RPY עבור הסימולציה השנייה למעבר בין מצבים	36
גרף 6.1 - התפלגות בינומילית כתלות במספר ההצלחות עבור הסתברויות שונות	72
גרף א.1 - זווית ה - Roll אשר נמדדה בזמן טיסה ידנית	96
גרף א.2 - זווית ה - Pitch אשר נמדדה בזמן טיסה ידנית	96
גרף א.3 - זווית ה - Yaw אשר נמדדה בזמן טיסה ידנית	97
גרף א.4 - הגובה הנמדד בזמן טיסה במצב אחיזת גובה	97
גרף א.5 - כוח הדחף המופעל כנגד מהירות סיבוב המנוע בריבוע	100
גרף א.6 - כוח הגרר המופעל כנגד מהירות סיבוב המנוע בריבוע	100
גרף א.7 - כוח הדחף המופעל כנגד אות הפיקוד הנשלח למנוע	101
גרף א.8 - כוח הגרר המופעל כנגד אות הפיקוד הנשלח למנוע	101
גרף א.9 - הזווית α כפונקציה של הזמן בכדי למצוא את I_{xx}	103
גרף א.10 - הזווית α כפונקציה של הזמן בכדי למצוא את I_{yy}	104
גרף א.11 - הזווית α כפונקציה של הזמן בכדי למצוא את I_{zz}	104
גרף א.12 - תוצאות התמרת פורייה של הגרף עבור I_{xx}	105
גרף א.13 - תוצאות התמרת פורייה של הגרף עבור I_{yy}	105
גרף א.14 - תוצאות התמרת פורייה של הגרף עבור I_{zz}	106

רשימת טבלאות

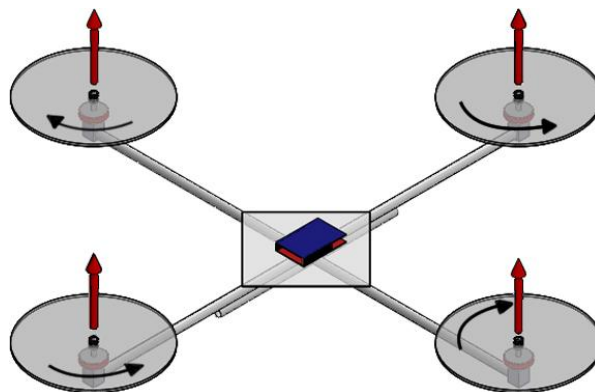
טבלה 6.1 - השוואת משכי הריצה וכמות האיטצוריות הדרושות עבור כל סימולציה	84
טבלה 6.2 - השוואת אורכי המסלולים הממוצעים המתקבלים	84
טבלה א.1 - תוצאות המדידות שהתקבלו בניסוי למציאת קבוע העילוי וקבוע הגרר	99
טבלה א.2 - כלל הקשרים שהתקבלו בניסוי למציאת קבוע העילוי וקבוע הגרר	102
טבלה א.3 - כלל הפרמטרים שנמדדו בכדי לחשב את מומנטי האינרציה של הכלי	106
טבלה א.4 - מומנטי האינרציה שהתקבלו עבור כלי הטיס	106

1 מבוא

ההתפתחות הטכנולוגית במהלך השנים האחרונות בתחומי החישה, עיבוד המידה והמיקרו-מעבדים איפשרה פיתוח וייצור של כלי טיס בלתי מאויישים (UAV - Unmanned Aerial Vehicles) - המוכרים יותר ככטבמ"ם. לכטבמ"ם שימושים רבים בעיקר בתחום הצבאי, לדוגמא ריגול, סריקה, נשיאת פצצות וכו'. בנוסף, במהלך השנים האחרונות לכטבמ"ם קיימים גם שימושים שונים בתחום האזרחי כגון צילומי משחקי כדורגל, צילומי נוף ופראצי, אבטחה ועוד.

את סוגי הכטבמ"ם השונים ניתן לסווג בצורות רבות. ראשית, ישנם כטבמ"ם הנשלטים ע"י אדם (באמצעות שלט לדוגמא), אך ישנם גם כאלה אוטונומיים המבצעים משימה מוגדרת מראש ללא הנחיית אדם. בנוסף, ניתן לחלק את הכטבמ"ם לכאלה בעלי כנפיים וכאלה בעלי רוטורים (לדוגמא מסוק). ההבדל המשמעותי בסיווג האחרון הוא שלכטבמ"ם בעלי רוטורים יש את היכולת להמריא או לנחות בצורה אנכית לגמרי (VTOL - Vertical Take-Off and Landing), ובנוסף הם בעלי יכולות תמרון גבוהות יותר. לכטבמ"ם בעלי כנפיים אין את היכולת הזו, אך לרוב הם בעלי יכולת לעמוד בטיסות ארוכות יותר, בטווחים ארוכים יותר ובמהירויות גבוהות יותר.

הקוואדרוטור (Quadrotor) הינו כטב"ם בעל רוטורים, כלומר VTOL בעל יכולות תמרון גבוהות במיוחד. בנוסף, ניתן לתכננו כך שיהיה אוטונומי לחלוטין. לקוואדרוטור ארבעה רוטורים (מכאן נגזר שמו) המסודרים בצורה של צלב מישורי. ארבעת הרוטורים מסודרים כך שכל רוטור יפעיל דחף כלפי מעלה, ובו בזמן כל זוג רוטורים סמוכים יסתובב בכיוונים מנוגדים (ראה איור 1.1). סידור כזה של הרוטורים מאפשר להפעיל מומנטים סביב שלושת הצירים הסטנדרטיים לצד הפעלת כוח כלפי מעלה, בניצב לצלב המישורי.



איור 1.1 : אוריינטציית הרוטורים ומבנה הקוואדרוטור.

היתרון הבולט ביותר של קוואדרוטור על פני מסוק (אליו נהוג להשוות אותו משום הדמיון הפיסי ביניהם) הוא שהוא איננו מסובך מבחינה מכאנית וכתוצאה מכך הוא זול בצורה משמעותית. בניגוד למסוק, זווית התקיפה של הרוטורים קבועה לכן אין צורך במנגנון מכאני יקר ומסובך, ואת השינוי בכיוון התנועה ניתן לקבל באמצעות הפעלת הרוטורים במהירויות סיבוב שונות.

יתרון נוסף (ומשמעותי במיוחד בעת ביצוע ניסויים) הוא כי בעצם החלוקה של הכוח לארבעה רוטורים, אנו גורמים לכך שעבור אותו כוח עילוי, לכל רוטור תהיה מהירות סיבוב נמוכה יותר וכתוצאה מכך גם אנרגיה קינטית קטנה יותר, ובכך בעת פגיעה של הרוטור הנזק יהיה קטן יותר.

מצד שני, לקוואדרוטור שש דרגות חופש אך ארבע כניסות (מהירויות הסיבוב של הרוטורים) בלבד, כלומר המערכת היא תת-ממונעת (Underactuated). בנוסף לכך, המערכת היא לא הולונומית (Nonholonomic), כלומר ישנו אילוץ (קשר) בין המהירויות והתאוצות. העובדה כי המערכת היא תת-ממונעת מקשה על השליטה והבקרה, בעוד העובדה כי המערכת היא לא הולונומית מאלצת תכנון של תנועה ולא תכנון של מסלול ואף מסבכת משמעותית את תכנון התנועה, וזאת משום שקיים צימוד בין היציאות - כלומר, לא ניתן להגיע לכל אוריינטציה או לכל מצב שאנו רוצים מפני שאין מספיק כניסות אשר ניתן להפעיל כך שנשלט בכל דרגות החופש.

מטרת עבודת מחקר זו היא בקרה ותכנון תנועה עבור קוואדרוטור. בכדי להשיג מטרה זו, נפתח ראשית את המודל הדינמי של הכלי. בנוסף, נציג חוק בקרה לא-ליניארי שיאפשר עקיבה אחרי מסלול מוגדר. לאחר מכן נציג שיטה לתכנון מסלול מנקודת מוצא נתונה לנקודת יעד רצויה תוך המנעות ממכשולים. בשלב הבא נציג ונפתח שיטה לתכנון תנועה ממצב מוצא נתון למצב יעד רצוי תוך המנעות ממכשולים במרחב עבודה מוכר מראש. לבסוף נבצע סימולציות, נציג את תהליך בניית כלי הטיס, כמו גם את מערכת הניסוי שהוקמה ונבצע ניסוי.

התרומות העיקריות של עבודת מחקר זו הינן מימוש אלגוריתם ה- RRT עבור קוואדרוטור בעל שנים-עשר משתני מצב תוך יישום של בקר לא-ליניארי, ושיפור אלגוריתם ה- RRT הקיים תוך התבססות על גרף וורונוי הכללי ויצירת אלגוריתם משופר לתכנון תנועה - RRTT, ומימושו עבור קוואדרוטור.

2 סקר ספרות

עבודות מחקר רבות ומחקרים רבים נערכו בנושא הקוואדרוטור. מרבית תחומי המפתח המשמשים למחקר בכל הנוגע לקוואדרוטור הינם בעיקר בניית מודל כלי הטיס, מדידה ושערוך של האוריינטציה והמיקום ופיתוח חוקי בקרה למיניהם. מספר המחקרים המתעסק בתכנון של מסלול ובמיוחד בתכנון של תנועה עבור הקוואדרוטור הוא קטן בצורה מפתיעה. סקר ספרות זה עוסק הן במודל ובבקרה של קוואדרוטור והן בשיטות שונות לתכנון מסלול/תנועה.

בין עבודות המחקר הנרחבות והמוכרות ביותר העוסקות בקוואדרוטור נמצאת עבודתו של Boubdallah [4] [5] [15] [16] [17] אשר עסק רבות בפיתוח מודל כלי הטיס, פיתוח בקרים לינאריים כגון LQ ו- PID ופיתוח בקרים לא לינאריים מבוססי Back-Stepping ו- Sliding-Mode. בעוד הבקרים אשר שימשו בעבודת מחקר זו הינם יישום של שיטות בקרה מוכרות עבור משוואות התנועה של הקוואדרוטור, במחקר שנערך ע"י Lee, Leok ו- McClamroch פותח בקר לא-לינארי אשר מותאם במיוחד לקוואדרוטור. בקר זה כולל מספר מצבי טיסה ופותר על-סמך שיטות בגיאומטריה דיפרנציאלית [2] [32].

בנוסף, פורסמו מחקרים כגון [33] ו- [34] אשר עוסקים בעיקר באפקטים האווירודינמיים הפועלים על הרוטורים ובהשפעה על כלי הטיס. האפקט האווירודינמי העיקרי שנבחן היה השפעת גמישות הרוטורים על מספר פרמטרים במהלך הטיסה. המסקנה העיקרית של מחקרים אילו היא כי ברוב המקרים, אפקטים אילו ניתנים להזנחה.

כאמור, תחום תכנון התנועה איננו תחום נפוץ יחסית לעבודות המחקר הנערכות בנושא הקוואדרוטור. מרבית המחקרים, כגון [2] [4] [5] [16] [17] מתעסקים בבקר אשר ייאפשר עקיבה אחר מסלול ולא בתכנון התנועה, או אפילו תכנון המסלול עצמו. למרות זאת, יש לציין את עבודתם [35] של Mellinger ו- Kumar אשר תכננו בקר ספציפי לעקיבה אחר מסלול ספציפי, ובנוסף תכננו את המסלול - אמנם לא ממש תכנון של מסלול אלא מעבר דרך מספר נקודות מוגדרות במרחב ע"ס הכרתו, אך התוצאות (בעיקר הסרטונים) מרשימות. בנוסף, קיימת עבודתם [36] של Hoffman ו- Waslander אשר מבוצע בה תכנון מסלול המבוסס על שיטת ה- Visibility Graph.

בתחום תכנון המסלול/תנועה קיימים ספרים רבים כמו [9] ו- [10] העוסקים בנושא זה בהרחבה רבה. בנוסף, מבוצעים מחקרים רבים עבור שיטות לתכנון שכזה. עבודת המחקר שבוצעה ע"י Kuffner ו- LaValle [3] [27] כוללת פיתוח של שיטת ה- RRT לתכנון תנועה במרחב המצב. אמנם ישנן דוגמאות למימוש השיטה עבור כלי טיס במרחב תלת-מימדי [3] [31], אך מדובר במערכות בעלות כניסות רבות יותר מן הקוואדרוטור. בנוסף, קיימות עבודות מחקר כדוגמת [30] אשר מתעסקת בתכנון תנועה במרחב תלת-מימדי עבור מטוסים וטיסנים.

בספרים [9] ו- [10] מוצגות שיטות רבות לתכנון של מסלול ביניהן נפוצות בעיקר שיטות ה- Generalized Voronoi Diagram, Visibility Graph ו- Potential Function. שיטות אילו מוכרות

וממומשות עבור מערכות רבות. שיטת גרף הראייה (Visibility Graph) הינה שיטה אשר ניתנת למימוש במרחבים דו או תלת מימדיים לכל היותר, מתבססת על מתיחת קווים ישרים בין "נקודות ראות" וטובה עבור מכשולים פוליגוניים בלבד. שיטה זו נפוצה בעיקר משום הפשטות והזריזות שבתכנון מסלול באמצעות שיטה זו ומשתמשים בה בעבודות רבות כדוגמת [36] [37]. חסרונה העיקר של שיטה זו נובע מן העובדה שהמסלול הנוצר יהיה צמוד למכשולים, כך שכלי הטיס יימצא בסכנה.

פונקציית הפוטנציאל הינה שיטה לתכנון תנועה המתבססת על בנייה של פונקצייה בהתאם למרחב העבודה/קונפיגורציה אשר תניב כניסות בקרה שייגרמו למערכת להתרחק מן המוצא והמכשולים אל עבר היעד. שיטה ממומשת פחות עבור כלי טיס וזאת משום שקיימת סחיפה (Drift), ובנוסף משום ששיטה זו למעשה מניחה שניתן להפעיל מספר כניסות כמספר דרגות החופש של המערכת. למרות זאת, ניתן בכל זאת לראות דוגמאות לעבודות שבהן השתמשו בפונקציית פוטנציאל עבור קוואדרוטור, לדוגמא [36].

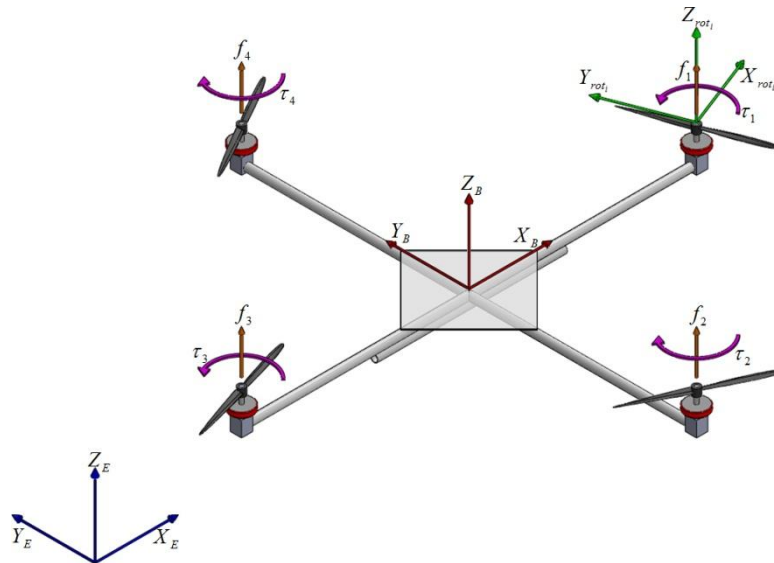
שיטת גרף וורונוי הכללי פחות נפוצה וזאת משום שהיא דורשת זמן רב יותר. מסיבה זו ישנן עבודות מחקר רבות כדוגמת [20] ו- [21] העוסקות בשיפור ובבנייה מהירה יותר של גרף וורונוי הכללי.

3 פיתוח משוואות הדינמיקה של הכלי

במהלך פרק זה נציג את הפיתוח המלא של משוואות התנועה של הקוואדרוטור באמצעות שיטת אויילר-לגראנז'. בכדי להבין פיתוח זה יש להכיר ראשית את המודל הדינמי והקינמטי של הקוואדרוטור. יש לציין כי בניגוד למרבית המחקרים העוסקים בנושא, פיתוח המשוואות הראשוני הוא מלא ונקי כמעט לגמרי מהזנחות.

3.1 המודל הדינמי של הקוואדרוטור

כל רוטור מונע ע"י מנוע (Brushless DC - BLDC במקרה שלנו), המפעיל מומנט על ציר הרוטור בכיוון הסיבוב של הרוטור. כתוצאה מהסיבוב ומהחיכוך שבין הרוטור לאוויר פועל מומנט נגדי על הרוטור וכתוצאה מכך גם על המנוע שמחובר לקוואדרוטור, ולכן המומנט פועל למעשה על הכלי עצמו. בנוסף כל רוטור מפעיל כוח דחף כלפי מעלה, כוח הפרופורציונלי למהירות הסיבוב שלו (ברביע). לכן נקבל בכל מקום בו מחובר מנוע לכלי הטיס כוח דחף כלפי מעלה, ומומנט בכיוון הפוך לכיוון סיבוב הרוטור המתאים (ראה איור 3.1). בנוסף לכך כתוצאה מכוח הכבידה נקבל כוח כלפי מטה, וההנחה היא כי הוא פועל במרכז הכובד שהוא גם המרכז הגיאומטרי של הכלי.



איור 3.1 : דיאגרמת הכוחות והמומנטים הפועלים על הכלי.

נניח כי כוח הדחף המתקבל מכל מנוע הוא מהצורה הבאה :

$$f_i = b\omega_i^2 \quad (1)$$

כאשר ω_i היא המהירות הזוויתית של המנוע ה- i , והפרמטר b הוא מקדם התלוי במספר גורמים כגון זווית התקיפה של הלהב, קבוע הגרר, צפיפות האוויר, טמפרטורת האוויר. ברור כי המקדם תלוי גם במהירות הסיבוב של הרוטור בצורה כלשהי, אך אנו נניח כי הוא קבוע וזהה עבור כל מנוע ומנוע.

אם נעתיק את כוחות הדחף של כל המנועים למרכז הכלי נקבל את כוח הדחף השקול, מִלֹּחַ במומנטים כתוצאה מההעתקה. כוח הדחף השקול הפועל במרכז הכלי בכיוון החיובי של ציר Z_b (כאשר האינדקס b מציין כי מדובר במערכת הגוף) נתון ע"י:

$$F_{tot} = f_1 + f_2 + f_3 + f_4 = b \cdot (\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \quad (2)$$

המומנטים הפועלים סביב מרכז הכלי כתוצאה מכוחות הדחף נתונים ע"י שתי המשוואות הבאות:

$$\tau_x = l \cdot f_4 - l \cdot f_2 = l \cdot b \cdot (\omega_4^2 - \omega_2^2) \quad (3)$$

$$\tau_y = l \cdot f_3 - l \cdot f_1 = l \cdot b \cdot (\omega_3^2 - \omega_1^2) \quad (4)$$

כאשר l מציין את אורך הזרוע בין מיקום המנוע לבין מרכז המסה של הכלי (אנו מניחים כי האורך זהה עבור ארבעת המנועים), והאינדקסים x, y מציינים את הציר במערכת הגוף סביבו פועל המומנט.

כפי שכבר ציינו, כתוצאה מהחיכוך שבין הרוטור לאוויר פועל מומנט על המנוע בכיוון הנגדי לכיוון הסיבוב, וכתוצאה מכך גם על כלי הטיס עצמו. המומנט המתקבל מכל מנוע הוא מהצורה הבאה:

$$\tau_i = d\omega_i^2 \quad (5)$$

וכתוצאה מארבעת המומנטים שמפעילים ארבעת המנועים נקבל כי המומנט הכללי הפועל בכיוון החיובי של ציר Z_b נתון ע"י:

$$\tau_z = \tau_4 + \tau_2 - \tau_3 - \tau_1 = d \cdot (\omega_4^2 + \omega_2^2 - \omega_3^2 - \omega_1^2) \quad (6)$$

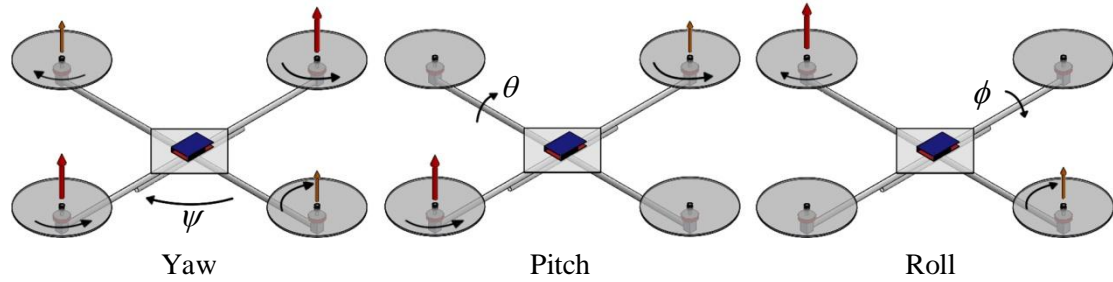
3.2 המודל הקינטי של הקוואדרוטור

פיתוח המודל הקינטי של הקוואדרוטור כולל מספר שלבים. ראשית נתאר מהן דרגות החופש של הכלי. בנוסף, נמצא את מטריצת הסיבוב ממערכת הגוף למערכת האינרציאלית ואת מטריצות הסיבוב מכל רוטור למערכת האינרציאלית. לאחר מכן נמצא את ווקטור המהירות הזוויתית של הכלי במערכת העולם.

3.2.1 דרגות החופש של הקוואדרוטור

לגוף הקוואדרוטור שש דרגות חופש, ולמערכת הכוללת את הרוטורים עשר דרגות חופש. לגוף הקוואדרוטור שלוש דרגות חופש המצינות את מיקום מרכז המסה של הכלי במערכת אינרציאלית, ושלוש דרגות חופש נוספות המצינות את האוריינטציה של הכלי במרחב. מיקום

מרכז המסה של הכלי נתון ע"י שלוש קואורדינטות במערכת אינרציאלית x_e, y_e, z_e , כאשר האינדקס e מציין כי הקואורדינטות הן ביחס למערכת צירים אינרציאלית. את שלושת דרגות החופש המציינות אוריינטציה ניתן לתאר באמצעות שלוש זוויות. זוויות הסיבוב סביב הצירים x, y, z נקראות (RPY) Roll, Pitch, Yaw ומסומנות ע"י ϕ, θ, ψ בהתאמה, כפי שניתן לראות באיור הבא:



איור 3.2 : תיאור זוויות אויילר ואוריינטצית גוף הכלי במרחב.

ארבעת דרגות החופש הנוספות של הכלי הן האוריינטציה של כל אחד מהרוטורים במרחב. כל רוטור חופשי להסתובב סביב ציר z במערכת הגוף ולכן זוויות הסיבוב של כל רוטור סביב ציר מתארות את ארבעת דרגות החופש האחרונות. נסמן את זווית הסיבוב של כל רוטור סביב ציר \hat{e}_i ב-

α_i .

3.2.2 מטריצות הסיבוב למערכת האינרציאלית

בכדי למצוא את מטריצות הסיבוב מהרוטורים וממערכת הגוף למערכת האינרציאלית, נמצא ראשית את מטריצת הסיבוב עבור סיבוב סביב כל ציר בנפרד.

מטריצת הסיבוב סביב ציר x נתונה ע"י:

$$Q_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix} \quad (7)$$

מטריצת הסיבוב סביב ציר y נתונה ע"י:

$$Q_y = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} \quad (8)$$

מטריצת הסיבוב סביב ציר z נתונה ע"י :

$$Q_z = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (9)$$

מטריצת הסיבוב ממערכת הגוף למערכת העולם תתקבל מהכפלה של שלושת המטריצות הללו לפי סדר הכפלה נתון. סדר ההכפלה בו נשתמש הוא $R = Q_z Q_y Q_x$. ביצוע מפורש של מכפלה זו ייתן :

$$R = \begin{bmatrix} \cos \psi \cos \theta & \cos \psi \sin \theta \sin \phi - \sin \psi \cos \phi & \sin \psi \sin \phi + \cos \psi \sin \theta \cos \phi \\ \sin \psi \cos \theta & \cos \psi \cos \phi + \sin \psi \sin \theta \sin \phi & \sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi \\ -\sin \theta & \cos \theta \sin \phi & \cos \theta \cos \phi \end{bmatrix} \quad (10)$$

בנוסף למטריצת הסיבוב ממערכת הגוף למערכת העולם, נרצה למצוא את מטריצת הסיבוב מכל רוטור למערכת העולם. בכדי לעשות זאת נמצא את מטריצת הסיבוב סביב ציר הסיבוב של כל רוטור.

מטריצת הסיבוב עבור סיבוב חיובי סביב ציר הסיבוב של הרוטור (כלומר עבור הרוטורים 1,3 נתונה ע"י :

$$Q_{ro\hat{t}_{1,3}} = \begin{bmatrix} \cos \alpha_i & -\sin \alpha_i & 0 \\ \sin \alpha_i & \cos \alpha_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (11)$$

ובאותו אופן, מטריצת הסיבוב עבור סיבוב שלילי סביב ציר הסיבוב של הרוטור (כלומר עבור הרוטורים 2,4) תהיה :

$$Q_{ro\hat{t}_{2,4}} = \begin{bmatrix} \cos \alpha_i & \sin \alpha_i & 0 \\ -\sin \alpha_i & \cos \alpha_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (12)$$

בכדי לקבל את מטריצת הסיבוב מרוטור כלשהו למערכת האינרציאלית נצטרך להכפיל את המטריצות לפי סדר הכפלה מסויים. מטריצת הסיבוב ממערכת הרוטור ה- i למערכת העולם נתונה ע"י :

$$R_{ro\hat{t}_i} = R \cdot Q_{ro\hat{t}_i} \quad (13)$$

3.2.3 ביטויים למהירות הזוויתית

עתה, לאחר שמצאנו את כל מטריצות הסיבוב, נוכל למצוא בקלות יחסית את ווקטורי המהירות הזוויתית של הגוף ושל כל רוטור במערכת האינרציאלית. בכדי למצוא את ווקטור המהירות הזוויתית של הכלי במערכת העולם נשתמש בזהות הבאה:

$$S(\omega) = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} = \dot{R}R^T \quad (14)$$

כאשר $\omega_x, \omega_y, \omega_z$ הן מהירויות הסיבוב של הכלי סביב הצירים x, y, z במערכת האינרציאלית בהתאמה.

לאחר חישוב אנליטי של נגזרת מטריצת הסיבוב, והכפלת המטריצות, ווקטור המהירות הזוויתית של הכלי במערכת העולם המתקבל הוא:

$$\bar{\omega}_{body} = \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} = \begin{pmatrix} \cos \psi \cos \theta \cdot \dot{\phi} - \sin \psi \cdot \dot{\theta} \\ \cos \psi \cdot \dot{\theta} + \sin \psi \cos \theta \cdot \dot{\phi} \\ \dot{\psi} - \sin \theta \cdot \dot{\phi} \end{pmatrix} \quad (15)$$

את הביטוי הנ"ל נוכל לרשום בצורה של מכפלת מטריצה בווקטור המהירות הזוויתית של הגוף (כלומר $\bar{\omega}_{body} = T_\omega \cdot [\dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T$), כאשר המטריצה מסמלת את מטריצת הטרנספורמציה בין צירי הסיבוב של הזוויות RPY לבין המערכת האינרציאלית. המטריצה הזו, אשר נקראת היעקביאן הרוטציוני, תראה כך:

$$T_\omega = \begin{bmatrix} \cos \psi \cos \theta & -\sin \psi & 0 \\ \sin \psi \cos \theta & \cos \psi & 0 \\ -\sin \theta & 0 & 1 \end{bmatrix} \quad (16)$$

כעת, נרצה לחשב גם את המהירות הזוויתית של הרוטורים ביחס למערכת האינרציאלית. בכדי לעשות זאת נשתמש בכלל הבא:

$$\bar{\omega}_{rot_e} = \bar{\omega}_{body_e} + \bar{\omega}_{rot_b} \quad (17)$$

כאשר $\bar{\omega}_{rot_e}$ הוא ווקטור המהירות הזוויתית של הרוטור במערכת העולם, $\bar{\omega}_{body_e}$ הוא ווקטור המהירות הזוויתית של הגוף במערכת העולם ו- $\bar{\omega}_{rot_b}$ הוא ווקטור המהירות הזוויתית של הרוטור ביחס למערכת הגוף. לאחר חיבור האיברים נקבל את הביטוי עבור המהירות הזוויתית של הרוטור במערכת העולם:

$$\bar{\omega}_{rot_i} = \begin{pmatrix} \cos \psi \cos \theta \cdot \dot{\phi} - \sin \psi \cdot \dot{\theta} + (-1)^{i+1} \dot{\alpha}_i (\sin \psi \sin \phi + \cos \psi \sin \theta \cos \phi) \\ \cos \psi \cdot \dot{\theta} + \sin \psi \cos \theta \cdot \dot{\phi} + (-1)^{i+1} \dot{\alpha}_i (\sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi) \\ \dot{\psi} - \sin \theta \cdot \dot{\phi} + (-1)^{i+1} \dot{\alpha}_i \cos \theta \cos \phi \end{pmatrix} \quad (18)$$

כאשר $\dot{\alpha}_i$ היא מהירות הסיבוב של הרוטור ה- i .

בנוסף לכך, נציין כי המהירות הקווית של הכלי נמדדת במערכת האינרציאלית, ונתונה ע"י:

$$\bar{v} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} \quad (19)$$

3.3 מציאת משוואות הדינמיקה באמצעות שיטת אוילר-לגראנז'

כעת, לאחר שמצאנו ביטויים למהירות הזוויתית של הכלי, למהירויות הזוויתיות של הרוטורים, למהירות הקווית של הכלי ולכוחות הפועלים עליו, נוכל לפתח את משוואות התנועה של הקואדרוטור באמצעות שיטת אוילר-לגראנז'.

לפני הפיתוח ישנן מספר הגדרות נוספת אותן נצטרך להגדיר. ראשית נגדיר את מטריצת האינרציה של הגוף במערכת הגוף, ואת מטריצת האינרציה של רוטור במערכת הרוטור. מטריצת האינרציה של הגוף נתונה ע"י:

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \quad (20)$$

כאשר I_{xx}, I_{yy}, I_{zz} מציינים את מומנטי האינרציה של הכלי סביב הצירים x_b, y_b, z_b בהתאמה. מטריצת האינרציה של כל רוטור במערכת הרוטור נתונה ע"י:

$$J = \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix} \quad (21)$$

כאשר J_{xx}, J_{yy}, J_{zz} מציינים את מומנטי האינרציה של הרוטור סביב הצירים $x_{rot}, y_{rot}, z_{rot}$ בהתאמה.

בנוסף להגדרות הללו נגדיר גם את ווקטור הקואורדינטות המוכללות בהן נשתמש :

$$q = \begin{pmatrix} x \\ y \\ z \\ \phi \\ \theta \\ \psi \end{pmatrix} \quad (22)$$

בכדי להשתמש בשיטת אויילר-לגראנז' יש למצוא ראשית ביטויים לאנרגיה הפוטנציאלית ולאנרגיה הקינטית של הכלי. ברור כי הביטוי עבור האנרגיה הפוטנציאלית של הכלי נובע מהגרביטציה בלבד ונתון ע"י :

$$V = mgz \quad (23)$$

כאשר m היא מסת הכלי, g - הוא קבוע הכבידה.

הביטוי עבור האנרגיה הקינטית של הגוף יתקבל מחיבור של האנרגיה הקינטית של הכלי כתוצאה ממהירות קווית, האנרגיה כתוצאה מסיבוב הגוף והאנרגיה מסיבוב ארבעת הרוטורים. הביטוי המתקבל הוא :

$$T = \frac{1}{2} m \bar{v} \bar{v}^T + \frac{1}{2} \bar{\omega}_{body}^T R I R^T \bar{\omega}_{body} + \frac{1}{2} \bar{\omega}_{rot1}^T R_{rot1} J R_{rot1}^T \bar{\omega}_{rot1} + \dots \quad (24)$$

$$\dots + \frac{1}{2} \bar{\omega}_{rot2}^T R_{rot2} J R_{rot2}^T \bar{\omega}_{rot2} + \frac{1}{2} \bar{\omega}_{rot3}^T R_{rot3} J R_{rot3}^T \bar{\omega}_{rot3} + \frac{1}{2} \bar{\omega}_{rot4}^T R_{rot4} J R_{rot4}^T \bar{\omega}_{rot4}$$

וכעת נוכל לקבל ביטוי עבור הלגראנגיאן (הפרש האנרגיות) :

$$L = T - V \quad (25)$$

כעת, בכדי למצוא את משוואות התנועה של הקואדרוטור נשתמש בשיטת אויילר-לגראנז'. לפי שיטה זו, מתקיים :

$$\frac{d}{dt} \left[\frac{\partial L}{\partial \dot{q}_i} \right] - \frac{\partial L}{\partial q_i} = \tau_{q_i} \quad (26)$$

כאשר q_i היא הקואורדינטה המוכללת ה- i , τ_{q_i} הוא הכוח הבלתי משמר בכיוון הקואורדינטה q_i . בשביל להשתמש במשוואה זו נצטרך למצוא את ווקטור הכוחות הבלתי משמרים בכיוון הקואורדינטות המוכללות. ראשית נצטרך להעביר את הכוחות ממערכת הגוף למערכת המרחב, כך שיהיו בכיוונים q_1, q_2, q_3 . נעשה זאת בצורה הבאה :

$$\begin{pmatrix} \tau_{q_1} \\ \tau_{q_2} \\ \tau_{q_3} \end{pmatrix} = R \begin{pmatrix} 0 \\ 0 \\ F_{tot} \end{pmatrix} \quad (27)$$

בנוסף, נצטרך להעביר את המומנטים ממערכת הגוף כך שיתאימו לקואורדינטות המוכללות שבחרנו, כלומר נצטרך להטיל את המומנטים על צירי הסיבוב של הזוויות RPY, כך שנקבל כוחות מוכללים בכיוונים q_4, q_5, q_6 . בכדי לעשות זאת נטיל תחילה את המומנטים ממערכת הגוף למערכת העולם, ולאחר מכן באמצעות היעקביאן הרוטציוני (16) המשוחלף נטיל אותם על צירי הסיבוב של זוויות ה-RPY:

$$\begin{pmatrix} \tau_{q_4} \\ \tau_{q_5} \\ \tau_{q_6} \end{pmatrix} = T_\omega^T R \begin{pmatrix} \tau_x \\ \tau_y \\ \tau_z \end{pmatrix} \quad (28)$$

לאחר מימוש השיטה עבור ששת הקואורדינטות המוכללות נקבל שש משוואות ובהן עשרת הפרמטרים המגדירים את דרגות החופש. ארבעת המשוואות הנוספות נובעות כתוצאה מהדינמיקה שבין הרוטור והמנוע. בכדי לתת ביטוי לדינמיקה זו נשתמש במודל עבור המנועים בהם השתמשנו (BLDC - Brushless DC). ע"פ מידול זה, נוכל לרשום כי:

$$\ddot{\alpha}_i = -\frac{k_m^2}{R_{res} J_{zz}} \dot{\alpha}_i - \frac{d}{\eta r^3 J_{zz}} \dot{\alpha}_i^2 + \frac{1}{k_m T_c} u_{current} \quad (29)$$

כאשר k_m הוא קבוע המומנט של המנוע, R_{res} ההתנגדות הפנימית של המנוע, d הוא אותו קבוע הגרר מהביטוי עבור המומנט (5), J_{zz} מומנט האינרציה של הרוטור סביב ציר הסיבוב שלו, T_c הינו קבוע הזמן של המנוע, r יחס התמסורת של הגיר במנוע, η נצילות הגיר ו- $u_{current}$ זרם הכניסה למנוע.

לסיכום, קיבלנו עשר משוואות המתארות את עשרת דרגות החופש של הכלי. המשוואות המתקבלות עצומות בגודלן ומסורבלות, ולכן נמנענו מרישום המשוואות.

3.4 הנחות ומעבר למודל פשוט

כעת ניקח את המשוואות שקיבלנו ונבצע מספר הנחות והזנחות בכדי לקבל מודל פשוט יותר ומשוואות תנועה פשוטות יותר איתן נוכל לעבוד, אשר מתארות נכונה את הדינמיקה של הכלי.

ההנחות שנניח עתה הן:

- זווית הסיבוב של הרוטור זניחה והאוריינטציה שלו זהה לזו של גוף הכלי.

- התאוצה הזוויתית של הרוטור זניחה.
- מומנט האינרציה של הרוטור קטן וזניח.
- הכניסות מעתה יהיו מהירויות הסיבוב של הרוטורים - או הכוחות שמפעילים הרוטורים.
- מדובר בזוויות קטנות.

ראשית ניתן להניח כי זווית הסיבוב של הרוטור זניחה וזאת משום שמהירות הסיבוב כה גבוהה שניתן להניח כי מדובר בדיסקה. בנוסף, מומנט האינרציה של הרוטור קטן בצורה משמעותית ממומנט האינרציה של גוף הכלי עצמו ולכן הוא זניח ביחס אליו. התאוצה הזוויתית של הרוטור קטנה יחסית ומוכפלת במספר גורמים קטנים ϕ - 1. לצורך נוחות נניח כי מהירויות הסיבוב (או הכוחות) של המנועים הן הכניסות ושניתן לספק את מהירויות הסיבוב (או את הכוחות) הדרושות (ע"י בקרה של המנועים בנפרד). בהנחת זוויות קטנות אנו מבטלים למעשה את כל האיברים של המהירות הזוויתית של הכלי שמופיעים בחזקה גדולה ϕ - 1, אך עדיין שומרים על הסינוסים והקוסינוסים, כך שלא מדובר בליניארזיה מוחלטת של המערכת - יש לציין כי הנחה זו לא תמיד מדויקת, אך ללא הנחה זו המשוואות ישארו עצומות ולא ניתן יהיה להשתמש בהן.

לאחר כל ההנחות הללו, המודל הפשוט יותר איתו נעבוד יהיה :

$$\begin{aligned}
 \ddot{x} &= (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi) \frac{1}{m} b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\
 \ddot{y} &= (\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi) \frac{1}{m} b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\
 \ddot{z} &= -g + (\cos \phi \cos \theta) \frac{1}{m} b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\
 \ddot{\phi} &= \dot{\theta} \dot{\psi} \left(\frac{I_{yy} - I_{zz}}{I_{xx}} \right) + \frac{l}{I_{xx}} b(\omega_4^2 - \omega_2^2) \\
 \ddot{\theta} &= \dot{\phi} \dot{\psi} \left(\frac{I_{zz} - I_{xx}}{I_{yy}} \right) + \frac{l}{I_{yy}} b(\omega_3^2 - \omega_1^2) \\
 \ddot{\psi} &= \dot{\phi} \dot{\theta} \left(\frac{I_{xx} - I_{yy}}{I_{zz}} \right) + \frac{1}{I_{zz}} d(\omega_2^2 + \omega_4^2 - \omega_1^2 - \omega_3^2)
 \end{aligned} \tag{30}$$

נוכל לבטא את מערכת המשוואות הנ"ל בצורה מטריציונית אשר תקל על העבודה בהמשך. נסמן

$$\Omega = \begin{bmatrix} \dot{\phi} & \dot{\theta} & \dot{\psi} \end{bmatrix}^T \text{ ונקבל:}$$

$$\begin{aligned}
 m\dot{\vec{v}} &= -mg \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} + R \begin{pmatrix} 0 \\ 0 \\ F_{tot} \end{pmatrix} \\
 I\dot{\Omega} + \Omega \times I\Omega &= \vec{M}
 \end{aligned} \tag{31}$$

4 בקרת עקיבה

בפרק זה נציג בקר לא-לינארי [2], אשר פותח עבור הקוואדרוטור ע"י Lee, Leok ו- McClamroch, בקר המאפשר עקיבה אחרי מסלול מוגדר. נניח לעת עתה כי אנו יודעים לתכנן מסלול, נתייחס אל פרמטרי המסלול הרצוי כידועים ונסמנם באינדקס d (מלשון desired - כלומר רצוי). בנוסף לכך, נוכיח את היציבות של הבקר. הרעיון שמאחורי פיתוח הבקר הוא כי קיימות ארבע כניסות בלבד, ולכן ניתן להביא לעקיבה אסימפטוטית אחרי ארבע דרגות חופש (מתוך שש במודל הפשוט). הבקר מחולק למעשה לשלושה מצבים אשר לכל אחד תפקיד שונה, ובעת ביצוע תמרון מסובך נחליף בין המצבים השונים תוך כדי טיסה. יתרונו הגדול ביותר של בקר זה הוא שלא בהכרח צריך להשתמש בזוויות אויילר (או RPY) בכדי לתאר אוריינטציה, ולכן הוא נמנע ממצבים סינגולריים שבהגדרה ע"י זוויות אויילר (לדוגמא כשהכלי הפוך). כמו כן, השגיאה ממנה הוא יכול להתחיל גדולה יחסית לבקרים אחרים. לצורך נוחות, נתייחס כעת אל הכניסות כאל כניסות של כוח ומומנטים, ולכן הכניסות למערכת יהיו:

$$\begin{pmatrix} F_{tot} \\ \tau_x \\ \tau_y \\ \tau_z \end{pmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & -l & 0 & l \\ -l & 0 & l & 0 \\ -\frac{d}{b} & \frac{d}{b} & -\frac{d}{b} & \frac{d}{b} \end{bmatrix} \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix} \quad (32)$$

בנוסף, נציג את אופרטור (העתקה למעשה) הכובע (\wedge) בו נשתמש רבות בפרק זה. אופרטור זה למעשה מדמה את פעולת הקרוס (Cross) בצורה מטריציונית:

$$\hat{x}y = x \times y \Rightarrow \hat{x} = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix} \quad (33)$$

כמו כן, נציג מספר מאפיינים נוספים של אופרטור זה, הניתנים להוכחה ע"י אלגברה פשוטה:

$$\hat{x}y = x \times y = -y \times x = -\hat{y}x \quad (34)$$

$$-\frac{1}{2} \text{trace}[\hat{x}\hat{y}] = x^T y \quad (35)$$

$$\text{trace}[\hat{x}A] = \text{trace}[A\hat{x}] = \frac{1}{2} \text{trace}[\hat{x}(A - A^T)] = -x^T (A - A^T)^v \quad (36)$$

$$\hat{x}A + A^T \hat{x} = (\{\text{trace}[A] \text{eye}(3) - A\}x)^\wedge \quad (37)$$

$$R\hat{x}R^T = (Rx)^\wedge \quad (38)$$

כאשר trace - סכום האיברים על האלכסון הראשי, ו- $\text{eye}(3)$ מטריצת יחידה בגודל 3×3 .

כעת נציג את שלושת מצבי הטיסה השונים של הבקר.

4.1 מצב עקיבת אוריינטציה

עבור מצב טיסה זה נציג בקר לא-ליניארי שמטרתו עקיבה אחרי אוריינטציה וגובה רצויים. בנוסף נוכיח כי בקר זה משיג עקיבה אסימפטוטית כמעט גלובאלית.

4.1.1 הגדרת בקר עבור מצב עקיבת אוריינטציה

ראשית נגדיר מספר הגדרות ולאחר מכן נציג את הבקר עבור מצב עקיבת אוריינטציה. נציין כי האוריינטציה הרצויה נתונה ע"י מטריצת הסיבוב R_d , והמהירות הזוויתית הרצויה של הגוף נתונה ע"י $\hat{\Omega}_{body} = R_d^T \dot{R}_d$. ראשית נגדיר את פונקציית השגיאה הנובעת מהבדלים בין האוריינטציה הרצויה לאוריינטציה הנתונה:

$$\Psi(R, R_d) = \frac{1}{2} \text{trace}(\text{eye}(3) - R_d^T R) \quad (39)$$

מהתכונות של מטריצות סיבוב ידוע כי $R^T R = \text{eye}(3)$, ולכן כאשר העקיבה תהיה מושלמת, כלומר נגיע למצב הרצוי בו $R = R_d$, פונקציית השגיאה תתאפס. כמו כן, נציין כי עבור הפרש של עד (לא כולל) 180° בין R ל- R_d מתקיים $\Psi(R, R_d) > 0$, כלומר הפונקצייה מוגדרת חיובית בתחום זה. נוכל להציג קבוצה זו ($\Psi(R, R_d) > 0$) אשר $R = R_d$ היא נקודת המינימום המקומית שלה) כתת קבוצה של Ψ בה מתקיים $\Psi(R, R_d) < 2$, כלומר עבור R_d נתון נוכל להגדיר את הקבוצה בצורה הבאה:

$$L_2 = \{R \in SO(3) \mid \Psi(R, R_d) < 2\} \quad (40)$$

כאשר $SO(3)$ (ראשי תיבות של Special Orthogonal Group) מייצגת את המטריצות המקיימות את התנאים הבאים: $\det(R) = 1$, $R^T R = \text{eye}(3)$. כלומר $SO(3)$ מייצגת את מרחב מטריצות הסיבוב במערכת צירים ימנית. תת הקבוצה L_2 שהגדרנו מכסה חלק נרחב מ- $SO(3)$.

לאחר גזירה של פונקציית השגיאה נקבל את הביטוי עבור שגיאת העקיבה אחרי האוריינטציה:

$$\bar{e}_R = \frac{1}{2} (R_d^T R - R^T R_d)^\vee \quad (41)$$

כאשר האופרטור $^\vee$ הפוך לאופרטור $^\wedge$ (ממטריצה skew סימטרית לוקטור). בנוסף, נרצה להגדיר את השגיאה של המהירות הזוויתית. בכדי לעשות זאת נרצה למעשה להשוות בין

הווקטורים המשיקים $\dot{R} \in T_R SO(3)$ ו- $\dot{R}_d \in T_{R_d} SO(3)$, כאשר $T_R SO(3)$ הוא המרחב המשיק ל- $SO(3)$ באוריינטציה R , ו- $T_{R_d} SO(3)$ הוא המרחב המשיק ל- $SO(3)$ באוריינטציה R_d . בכדי להשוות בין הווקטורים המשיקים יש לבצע טרנספורמציה לאחד מהם מפני שהם שוכנים במרחבים משיקים שונים. נעשה זאת ע"י העתקה מהמרחב המשיק $T_R SO(3)$ למרחב המשיק $T_{R_d} SO(3)$:

$$\dot{R} - \dot{R}_d (R_d^T R) = R (\Omega - R^T R_d \Omega_d)^\wedge \quad (42)$$

נבחר את שגיאת המהירות הזוויתית להיות האיבר שבסוגריים:

$$\bar{e}_\Omega = \Omega - R^T R_d \Omega_d \quad (43)$$

בנוסף, נראה כי שגיאת המהירות הזוויתית היא בעצם המהירות הזוויתית של מטריצת הסיבוב היחסית $R_d^T R$, אשר מייצגת מעבר בין האוריינטציה R לאוריינטציה R_d^T :

$$\frac{d}{dt} (R_d^T R) = \dot{R}_d^T R + R_d^T \dot{R} \quad (44)$$

נשתמש בעובדות הבאות: $\dot{R} = R \hat{\Omega}$ ו- $\dot{R}_d = R_d \hat{\Omega}_d$, ובנוסף $\hat{\Omega}^T = -\hat{\Omega}$ (מטריצה skew סימטרית). נציב ונקבל:

$$\frac{d}{dt} (R_d^T R) = -\hat{\Omega}_d^T R_d^T R + R_d^T R \hat{\Omega} \quad (45)$$

בעזרת פעולות מתמטיות פשוטות (הוצאת גורם משותף) נקבל את הביטוי הבא:

$$\frac{d}{dt} (R_d^T R) = R_d^T R (\hat{\Omega} - R^T R_d \hat{\Omega}_d R_d^T R) \quad (46)$$

נשתמש במשוואה (38) ונחליף את הביטוי $R^T R_d \hat{\Omega}_d R_d^T R$ בביטוי $(R^T R_d \Omega_d)^\wedge$ ונקבל:

$$\frac{d}{dt} (R_d^T R) = R_d^T R (\hat{\Omega} - (R^T R_d \Omega_d)^\wedge) \quad (47)$$

וכעת לפי משוואה (43) נוכל לרשום:

$$\frac{d}{dt} (R_d^T R) = (R_d^T R) \hat{e}_\Omega \quad (48)$$

כלומר, שגיאת המהירות הזוויתית שנבחרה היא המהירות הזוויתית של מטריצת הסיבוב היחסית, לכן השגיאה שנבחרה תתן אינדיקציה טובה לגבי קצב ההתקרבות (או ההתרחקות) בין האוריינטציה הנתונה לזו הרצויה.

כעת, נבחר (ע"פ הדינמיקה של הכלי) את ווקטור הכניסה של המומנטים להיות :

$$M = -k_R \bar{e}_R - k_\Omega \bar{e}_\Omega + \Omega \times I \Omega - I(\hat{\Omega} R^T R_d \Omega_d - R^T R_d \dot{\Omega}_d) \quad (49)$$

כאשר k_R, k_Ω הם הגברים חיוביים כלשהם. נציין כעת (ונוכיח זאת בהמשך) כי הנקודה $(e_\Omega, e_R) = (\bar{0}, \bar{0})$ היא נקודת שיווי משקל יציבה אקספוננציאלית, כאשר תנאי ההתחלה מקיימים :

$$\Psi(R(0), R_d(0)) < 2 \quad (50)$$

$$\|e_\Omega(0)\|^2 < \frac{2}{\lambda_{\max}(I)} k_R (2 - \Psi(R(0), R_d(0))) \quad (51)$$

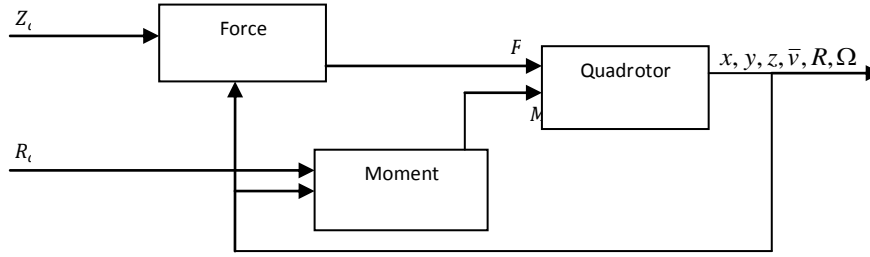
כאשר $\lambda_{\max}(I)$ הוא הערך העצמי המקסימלי של מטריצת האינרציה של הגוף, ו- $\|e_\Omega(0)\|$ הגודל של ווקטור שגיאת המהירות הזוויתית ההתחלתית. מלבד החשיבות שבהצגת תנאי ההתחלה בצורה מתמטית, ניתן להסביר את תנאי ההתחלה הללו בצורה אינטואיטיבית יותר. התנאי הראשון (50), מבטיח שהנקודה $R = R_d$ תהיה נקודת המינימום המקומית של תחום העבודה בו אנו נמצאים, ושההתכנסות (או העקיבה) תוביל למצב הרצוי. התנאי השני (51) מבטיח שקצב ההתרחקות מהאוריינטציה הרצויה קטן מספיק כך שניתן יהיה לשלוט בשינוי זה לפני התרחקות גדולה מדי, ויציאה מתחום העבודה שבו $R = R_d$ היא נקודת המינימום המקומית.

עד עתה השתמשנו בשלוש כניסות בכדי לטפל בשלושת דרגות החופש של האוריינטציה. ננצל את הכניסה האחרונה בכדי להשיג עקיבה אקספוננציאלית אחרי גובה הכלי במרחב (דהיינו z) :

$$F_{tot} = \frac{-k_z(z - z_d) - k_v(\dot{z} - \dot{z}_d) + mg + m\ddot{z}_d}{e_3 \cdot \text{Re}_3} \quad (52)$$

כאשר z_d הוא המסלול הרצוי לגובה, k_z, k_v הגברים חיוביים כלשהם ו- $e_3 = (0, 0, 1)^T$. אנו מניחים כי $e_3 \cdot \text{Re}_3 \neq 0$.

דיאגרמת הבלוקים עבור מצב טיסה זה תהיה:



איור 4.1: דיאגרמת הבלוקים עבור מצב בקרת אוריינטציה.

4.1.2 הוכחת יציבות עבור מצב עקיבת אוריינטציה

כעת נוכיח כי $(e_\Omega, e_R) = (\bar{0}, \bar{0})$ היא נקודת שיווי משקל יציבה אקספוננציאלית עבור תנאי ההתחלה שצויינו בסעיף הקודם. בכדי לעשות זאת נמצא ראשית את הדינמיקה של השגיאות שהגדרנו, כלומר נמצא את $\dot{e}_R, \dot{e}_\Omega$, ובנוסף נגדיר פונקציית ליאפונוב מתאימה. לאחר מכן נראה שתחת התנאים שצויינו (משוואות (51), (50)) $R(t)$ יימצא תמיד בתת-הקבוצה L_2 , וכתוצאה מכך $\Psi(R, R_d) > 0$ סביב $R = R_d$, ומכאן נראה יציבות אקספוננציאלית.

ראשית נרצה למצוא את הדינמיקה של השגיאות. בכדי לעשות זאת נגזור את Ψ לפי הזמן, ונקבל:

$$\dot{\Psi}(R, R_d) = -\frac{1}{2} \text{trace} \left(\frac{d}{dt} (R_d^T R) \right) \quad (53)$$

נשתמש במשוואה (48) ונקבל:

$$\dot{\Psi}(R, R_d) = -\frac{1}{2} \text{trace} (R_d^T R \hat{e}_\Omega) \quad (54)$$

כעת, בעזרת משוואה (36) נוכל לקבל:

$$\dot{\Psi}(R, R_d) = \frac{1}{2} e_\Omega^T (R_d^T R - R^T R_d)^\vee \quad (55)$$

וכעת לפי (41) נקבל:

$$\dot{\Psi}(R, R_d) = e_\Omega^T e_R = e_R \cdot e_\Omega \quad (56)$$

כעת נמצא את הביטוי עבור \dot{e}_R . נגזור את (41) לפי הזמן:

$$\dot{e}_R = \frac{1}{2} \left(\frac{d}{dt} (R_d^T R) - \frac{d}{dt} (R^T R_d) \right)^\vee \quad (57)$$

נשתמש ב- (48) ונקבל :

$$\dot{e}_R = \frac{1}{2} (R_d^T R \hat{e}_\Omega + \hat{e}_\Omega R^T R_d)^\vee \quad (58)$$

בעזרת (37) נקבל כי :

$$\dot{e}_R = \frac{1}{2} (\text{trace}[R^T R_d] \text{eye}(3) - R^T R_d) e_\Omega \equiv C(R^T R_d) e_\Omega \quad (59)$$

כעת נרצה להראות כי $\|C(R^T R_d)\|_2 \leq 1$ לכל $R^T R_d \in SO(3)$, כאשר $\|A\|_2 = \sqrt{\lambda_{\max}(A^* A)}$ (נורמה אוקלידית למטריצה). כלומר אנו מחפשים את הערכים העצמיים של $C(R^T R_d)$ ו- $C^T(R^T R_d)$. ידוע כי ניתן לייצג מטריצת סיבוב כאקספוננט של מטריצה skew סימטרית, כלומר $R^T R_d = \exp(\hat{\omega})$. לפי נוסחאת אוילר-רודריגז ניתן להראות שהערכים העצמיים של $C^T(\exp(\hat{\omega}))C(\exp(\hat{\omega}))$ נתונים ע"י $(\frac{1}{2}(1 + c\|\omega\|), \frac{1}{2}(1 + c\|\omega\|), \frac{1}{2}(1 + c\|\omega\|))$. כלומר הערכים העצמיים קטנים או שווים ל-1 עבור כל $\omega \in R^3$, ולכן $\|C(R^T R_d)\|_2 \leq 1$. כעת נוכל לרשום (לפי (59) כי :

$$\|\dot{e}_R\| \leq \|e_\Omega\| \quad (60)$$

נגזור את הביטוי עבור \bar{e}_Ω (משוואה (43)) ונכפיל במטריצת האינרציה I משמאל :

$$I\dot{e}_\Omega = I\dot{\Omega} - IR^T R_d \dot{\Omega}_d - IR^T R_d \hat{\Omega}_d \Omega_d + I\hat{\Omega} R^T R_d \Omega_d \quad (61)$$

האיבר $IR^T R_d \hat{\Omega}_d \Omega_d$ (האיבר השלישי באגף הימני של המשוואה) מתאפס מפני ש- $\hat{\Omega}_d \Omega_d = \Omega_d \times \Omega_d = 0$ ונקבל כי :

$$I\dot{e}_\Omega = I\dot{\Omega} + I(\hat{\Omega} R^T R_d \Omega_d - R^T R_d \dot{\Omega}_d) \quad (62)$$

נשלב את המשוואות עבור מומנט הבקרה הנבחר (49) והדינמיקה של המערכת (31) ובאמצעות פעולות פשוטות נקבל :

$$I\dot{\Omega} + I(\hat{\Omega} R^T R_d \Omega_d - R^T R_d \dot{\Omega}_d) = -k_R e_R - k_\Omega e_\Omega \quad (63)$$

נציב את (62) לתוך (63) ונקבל כי :

$$I\dot{e}_\Omega = -k_R e_R - k_\Omega e_\Omega \quad (64)$$

כעת, לאחר שמצאנו את הדינמיקה של השגיאות נוכל להגדיר פונקציית ליאפונוב. נבחר פונקציית ליאפונוב מהצורה הבאה :

$$v_1 = \frac{1}{2} e_\Omega \cdot I e_\Omega + k_R \Psi(R, R_d) + c_1 e_R \cdot e_\Omega \quad (65)$$

כאשר c_1 קבוע לא שלילי כלשהו.

נזכיר כי בכדי להוכיח יציבות אקספוננציאלית לפי ליאפונוב יש להראות כי מתקיים :

$$\begin{aligned} k_1 \|\bar{x}\|^a &\leq v_1(z) \leq k_2 \|\bar{x}\|^a \\ \dot{v}_1(z) &\leq -k_3 \|\bar{x}\|^a \end{aligned} \quad (66)$$

לכל \bar{x} , כאשר k_1, k_2, k_3, a קבועים חיוביים כלשהם ו- \bar{x} ווקטור משתני המצב. בכדי להראות זאת, נגזור את פונקציית ליאפונוב שנבחרה (65) לפי הזמן :

$$\dot{v}_1 = \frac{1}{2} \dot{e}_\Omega \cdot I e_\Omega + \frac{1}{2} e_\Omega \cdot I \dot{e}_\Omega + k_R \dot{\Psi}(R, R_d) + c_1 \dot{e}_R \cdot e_\Omega + c_1 e_R \cdot \dot{e}_\Omega \quad (67)$$

כעת נציב את הביטויים עבור הנגזרות כפי שקיבלנו במשוואות (56), (59), (64) :

$$\begin{aligned} \dot{v}_1 &= e_\Omega \cdot II^{-1}(-k_R e_R - k_\Omega e_\Omega) + k_R e_R \cdot e_\Omega + c_1 C(R^T R) e_\Omega \cdot e_\Omega + \dots \\ &\dots + c_1 e_R \cdot I^{-1}(-k_R e_R - k_\Omega e_\Omega) \end{aligned} \quad (68)$$

וע"י פעולות פשוטות נוכל לקבל את הביטוי הבא :

$$\dot{v}_1 = -k_\Omega \|e_\Omega\|^2 + c_1 C(R^T R) \|e_\Omega\|^2 - k_R c_1 e_R \cdot I^{-1} e_R - k_\Omega c_1 e_R \cdot I^{-1} e_\Omega \quad (69)$$

כפי שראינו (ממשוואה 60) מתקיים $\|C(R^T R)\|_2 \leq 1$ ולכן הנגזרת של פונקציית ליאפונוב תהיה חסומה ע"י :

$$\dot{v}_1 \leq -(k_\Omega - c_1) \|e_\Omega\|^2 - \frac{k_R c_1}{\lambda_{\max}(I)} \|e_R\|^2 + \frac{k_\Omega c_1}{\lambda_{\min}(I)} \|e_R\| \|e_\Omega\| \quad (70)$$

ובצורה מטריציונית נוכל לרשום כי :

$$\dot{v}_1 \leq -z_1^T W_1 z_1 \quad (71)$$

כאשר $z_1^T = \begin{bmatrix} \|e_R\| & \|e_\Omega\| \end{bmatrix}$ ו- W_1 נתון ע"י :

$$W_1 = \begin{bmatrix} \frac{k_R c_1}{\lambda_{\max}(I)} & -\frac{k_\Omega c_1}{2\lambda_{\min}(I)} \\ -\frac{k_\Omega c_1}{2\lambda_{\min}(I)} & k_\Omega - c_1 \end{bmatrix} \quad (72)$$

עתה, לאחר שמאצנו ביטוי לחסם של הנגזרת של פונקציית ליאפונוב, נחפש ביטוי לחסמים של הפונקציה עצמה.

נגדיר תחילה $\nu_1' = \nu_1|_{c_1=0}$, ולפי המשוואות (65), (70) נוכל לקבל כי :

$$\begin{aligned} \nu_1' &= \frac{1}{2} e_\Omega \cdot I e_\Omega + k_R \Psi(R, R_d) \\ \dot{\nu}_1' &= -k_\Omega \|e_\Omega\|^2 \leq 0 \end{aligned} \quad (73)$$

ניתן לראות כי הנגזרת של פונקציית ליאפונוב שלילית למחצה, בעוד הפונקצייה עצמה חיובית למחצה (בתחום הנתון לפי התנאי (50)), כלומר הערך של הפונקציה יהיה קטן, או לכל היותר שווה, מהערך של הפונקציה בזמן ההתחלתי, כלומר: $\nu_1'(t) \leq \nu_1'(0)$. כמו כן, ניתן לראות כי הערך של פונקציית ליאפונוב שהגדרנו חסום ע"י :

$$\nu_1' \leq \frac{1}{2} \|e_\Omega\|^2 \lambda_{\max}(I) + k_R \Psi(R, R_d) \quad (74)$$

כעת, אם נציב את התנאי שהגדרנו במשוואה (51), קל לקבל כי :

$$\nu_1'(t) \leq \nu_1'(0) < 2k_R \quad (75)$$

בנוסף, לפי משוואה (74) נוכל לרשום כי מתקיים :

$$k_R \Psi(R, R_d) \leq \nu_1'(t) \quad (76)$$

וכעת מחיבור של התנאים שהתקבלו בשתי המשוואות האחרונות נקבל כי :

$$k_R \Psi(R, R_d) \leq \nu_1'(t) \leq \nu_1'(0) < 2k_R \quad (77)$$

מכאן, נוכל לקבל כי קיים קבוע המקיים את התנאי הבא :

$$\Psi(R(t), R_d(t)) \leq \gamma_1 < 2 \quad (78)$$

כלומר, פונקציית השגיאה $\Psi(R(t), R_d(t))$ שוכנת בתת-הקבוצה L_{γ_1} ששייכת לקבוצה L_2

($L_{\gamma_1} \subset L_2$), כפי שהגדרנו ב- (40), ולכן מתקיים $\Psi(R, R_d) > 0$ סביב $R = R_d$.

נתבונן בנוסחאת אויילר-רודריגז (שכבר הזכרנו) :

$$\exp(\hat{\omega}) = \text{eye}(3) + \frac{\hat{\omega}}{\|\omega\|} \sin\|\omega\| + \frac{\hat{\omega}^2}{\|\omega\|^2} (1 - \cos\|\omega\|) \quad (79)$$

נסמן $R^T R_d = \exp(\hat{\omega})$ ונציב את הביטוי שקיבלנו ב- (79) לתוך הביטוי עבור $\Psi(R(t), R_d(t))$ שהגדרנו במשוואה (39) נקבל כי :

$$\Psi(R(t), R_d(t)) = 1 - \cos\|\omega\| \quad (80)$$

אם נשתמש פעם נוספת במשוואה (79), אך נציב את הביטוי לתוך (41) נקבל כי מתקיים :

$$\|e_R\|^2 = \sin^2\|\omega\| = (1 - \cos\|\omega\|)(1 + \cos\|\omega\|) = (1 + \cos\|\omega\|) \cdot \Psi(R(t), R_d(t)) \quad (81)$$

ע"י שימוש במשוואות של (80) ו- (81) והעברת אגפים נוכל לקבל :

$$\Psi(R(t), R_d(t)) = \frac{\|e_R\|^2}{2 - \Psi(R(t), R_d(t))} \quad (82)$$

ובהצבת החסמים שקיבלנו ב- (78), נקבל כי :

$$\frac{\|e_R\|^2}{2} \leq \Psi(R(t), R_d(t)) \leq \frac{\|e_R\|^2}{2 - \gamma_1} \quad (83)$$

וכעת, אם נציב חסמים אילו לתוך הביטוי עבור הגזרת של פונקציית ליאפנוב (65) נוכל לקבל כי פונקציית ליאפנוב חסומה ע"י :

$$\frac{1}{2} \lambda_{\min}(I) \|e_\Omega\|^2 + k_R \frac{\|e_R\|^2}{2} - c_1 \|e_R\| \|e_\Omega\| \leq \nu_1 \leq \frac{1}{2} \lambda_{\max}(I) \|e_\Omega\|^2 + k_R \frac{\|e_R\|^2}{2 - \gamma_1} + c_1 \|e_R\| \|e_\Omega\| \quad (84)$$

ובצורה מטריציונית נוכל לרשום כי :

$$z_1^T M_{11} z_1 \leq \nu_1 \leq z_1^T M_{12} z_1 \quad (85)$$

כאשר $z_1^T = [\|e_R\| \quad \|e_\Omega\|]$, והמטריצות M_{11}, M_{12} נתונות ע"י :

$$M_{11} = \frac{1}{2} \begin{bmatrix} k_R & -c_1 \\ -c_1 & \lambda_{\min}(I) \end{bmatrix}, M_{12} = \frac{1}{2} \begin{bmatrix} \frac{2k_R}{2-\gamma_1} & c_1 \\ c_1 & \lambda_{\max}(I) \end{bmatrix} \quad (86)$$

בכדי להוכיח יציבות אקספוננציאלית נצטרך לבחור את הקבוע c_1 כך שהמטריצות שהגדרנו ב- (86) ו- (72) יהיו חיוביות, וזאת בכדי שהערכים העצמיים של המטריצות יהיו חיוביים גם כן. בחירה של c_1 בצורה הבאה תבטיח חיוביות של שלושת המטריצות הללו:

$$c_1 < \min \left\{ k_\Omega, \frac{4k_\Omega k_R \lambda_{\min}(I)^2}{k_\Omega^2 \lambda_{\max}(I) + 4k_R \lambda_{\min}(I)^2}, \sqrt{k_R \lambda_{\min}(I)} \right\} \quad (87)$$

ולסיכום, נוכל כעת לרשום כי:

$$\begin{aligned} \lambda_{\min}(M_{11}) \|z_1\|^2 &\leq \dot{v}_1 \leq \lambda_{\max}(M_{12}) \|z_1\|^2 \\ \dot{v}_1 &\leq -\lambda_{\min}(W_1) \|z_1\|^2 \end{aligned} \quad (88)$$

כלומר, קיבלנו כי עבור תנאי ההתחלה שהגדרנו ב- (50) ו- (51) נקבל כי הנקודה $(e_\Omega, e_R) = (\bar{0}, \bar{0})$ היא אכן נקודת שיווי משקל יציבה אקספוננציאלית ולכן יש עקיבה אחרי אוריינטציה רצויה.

בכדי להראות עקיבה אסימפטוטית אחרי הגובה הרצוי נכפול סקלארית את משוואות התנועה (31) בווקטור יחידה בכיוון z_e ונקבל:

$$m\ddot{z} = -mg + F_{tot} \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T \cdot R \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T \quad (89)$$

ע"י הצבת הביטוי עבור הכוח ממשוואה (52) נקבל כי:

$$m(\ddot{z} - \ddot{z}_d) = -k_x(z - z_d) - k_v(\dot{z} - \dot{z}_d) \quad (90)$$

ברור מצורת המשוואה (בדומה לניתוח של מערכת *SISO* פשוטה) כי מתקבלת עקיבה אקספוננציאלית אחרי הגובה הרצוי.

4.2 מצב עקיבת מיקום

עבור מצב טיסה זה נציג בקר לא-לינארי המשיג עקיבה אסימפטוטית כמעט גלובאלית אחרי מיקום רצוי ושואף לעקיבה אחרי אוריינטציה רצויה.

4.2.1 הגדרת בקר עבור מצב עקיבת מיקום

בכדי להגיע לעקיבה אחרי מיקום רצוי נרצה להפעיל כוח בכיוון ספציפי. בכדי להשיג זאת נצטרך לשנות את האוריינטציה של הכלי במרחב וזאת משום שניתן להפעיל כוח בכיוון ציר z של הכלי בלבד. לכן נגדיר מטריצת סיבוב שתביא את הכלי למצב בו הכוח פועל בכיוון הרצוי. תחילה נגדיר את שגיאות המיקום עבור מצב טיסה זה:

$$e_{\bar{x}} = \bar{x} - \bar{x}_d \quad (91)$$

כאשר \bar{x} ו- \bar{x}_d הינם ווקטור המיקום וווקטור הרצוי בהתאמה. שגיאת המהירות עבור מצב טיסה זה תהיה:

$$e_v = \bar{v} - \dot{\bar{x}}_d \quad (92)$$

כאשר \bar{v} ו- $\dot{\bar{x}}_d$ הם ווקטור המהירות הרצויה וווקטור המהירות בהתאמה. נגדיר את כניסת כוח הדחף להיות:

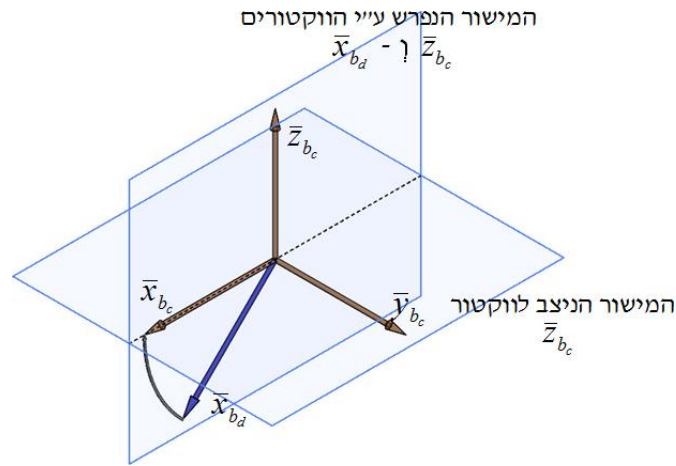
$$F_{tot} = (-k_x e_{\bar{x}} - k_v e_v + m g e_3 + m \ddot{\bar{x}}_d) \cdot R \cdot e_3 \quad (93)$$

כאשר k_x ו- k_v הם הגברים חיוביים כלשהם. בכדי לכוון את כלי הטיס כך שכוח הדחף יפעל בכיוון הרצוי, נרצה לכוון את הווקטור \bar{z}_{b_c} (כאשר האינדקס c הוא מהמילה controlled, כלומר הנשלט) בצורה הבאה:

$$\bar{z}_{b_c} = \frac{-k_x e_{\bar{x}} - k_v e_v + m g e_3 + m \ddot{\bar{x}}_d}{\| -k_x e_{\bar{x}} - k_v e_v + m g e_3 + m \ddot{\bar{x}}_d \|} \quad (94)$$

כאשר אנו מניחים כי: $\| -k_x e_{\bar{x}} - k_v e_v + m g e_3 + m \ddot{\bar{x}}_d \| \neq 0$, ובנוסף כי התאוצה הרצויה חסומה כך ש: $\| m g e_3 + m \ddot{\bar{x}}_d \| < B$ עבור קבוע חיובי B .

בכדי להגדיר מטריצת סיבוב רצויה אנו צריכים להגדיר שני ווקטורים רצויים נוספים כך שיהיו אורתוגונליים ויקיימו את התנאים עבור מטריצת סיבוב. בכדי לעשות זאת נבחר את הווקטור \bar{x}_{b_c} להיות ניצב לווקטור \bar{z}_{b_c} - אילוץ הכרחי הנובע ממערכת צירי הגוף שנבחרה (איור 3.1). בנוסף לכך יש לנו עדיין חופש כיצד לבחור את \bar{x}_{b_c} , כלומר לבחור את כיוונו על המישור הניצב לווקטור \bar{z}_{b_c} , ולכן נבחר אותו כך שהוא ייתלכד עם היטל הווקטור הרצוי \bar{x}_{b_d} על המישור הניצב ל- \bar{z}_{b_c} וכך נשיג עקיבה אחרי פרמטר של אוריינטציה (ראה איור 4.2).



איור 4.2: דרך בחירת הווקטור \bar{x}_{b_c} .

בצורה מתמטית, נוכל להגדיר את הווקטור \bar{x}_{b_c} בצורה הבאה (תחת ההנחה כי \bar{x}_{b_d} ו- \bar{z}_{b_c} אינם ניצבים):

$$\bar{x}_{b_c} = -\frac{1}{\|\bar{z}_{b_c} \times \bar{x}_{b_d}\|} \left(\bar{z}_{b_c} \times (\bar{z}_{b_c} \times \bar{x}_{b_d}) \right) \quad (95)$$

כאשר אנו מניחים כי $\|\bar{z}_{b_c} \times \bar{x}_{b_d}\| \neq 0$. מאחר ויש לנו את שני הווקטורים הרצויים \bar{x}_{b_d} ו- \bar{z}_{b_c} נוכל לקבל בקלות את הווקטור השלישי בצורה הבאה:

$$\bar{y}_{b_c} = \bar{z}_{b_c} \times \bar{x}_{b_c} \quad (96)$$

וכעת, לאחר שמצאנו את שלושת הווקטורים הרצויים נוכל להגדיר את מטריצת הרוטציה הרצויה בצורה הבאה:

$$R_c = [\bar{x}_{b_c}, \bar{y}_{b_c}, \bar{z}_{b_c}] \quad (97)$$

למעשה, במצב טיסה זה (עקיבת מיקום), המטרה היא להגיע למיקום רצוי ע"י בחירת אוריינטציה רצויה שתוביל למיקום הרצוי ועקיבה אחרי אוריינטציה רצויה זו. בכדי להשיג עקיבה אחרי האוריינטציה הרצויה (שבסופו של דבר תביא למיקום רצוי) נבנה את ווקטור כניסת המומנטים בצורה זהה למצב בקרת אוריינטציה, אך נחליף במשוואות (49), (43), (41) את הביטוי R_d בביטוי R_c . כלומר השגיאות עבור האוריינטציה והמהירות הזוויתית יראו כעת כך:

$$e_R = \frac{1}{2} (R_c^T R - R^T R_c)^\vee \quad (98)$$

$$e_\Omega = \Omega - R^T R_c \Omega_c \quad (99)$$

וכעת, נגדיר את ווקטור כניסת המומנטים להיות:

$$M = -k_R e_R - k_\Omega e_\Omega + \Omega \times I \Omega - I(\hat{\Omega} R^T R_c \Omega_c - R^T R_c \dot{\Omega}_c) \quad (100)$$

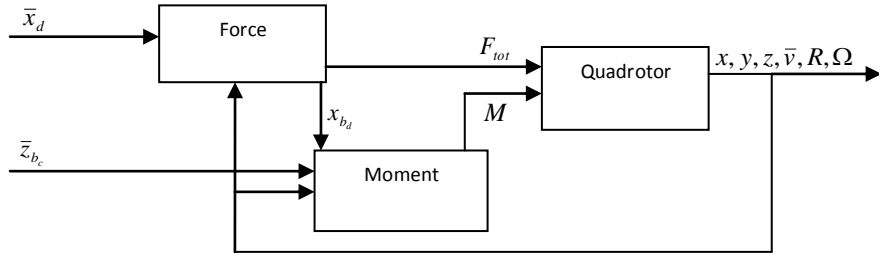
הנקודה $(\bar{e}_x, \bar{e}_y, \bar{e}_\Omega, \bar{e}_R) = (\bar{0}, \bar{0}, \bar{0}, \bar{0})$ היא נקודת שיווי משקל יציבה אקספוננציאלית, כאשר תנאי ההתחלה מקיימים:

$$\Psi(R(0), R_c(0)) < 1 \quad (101)$$

$$\|e_\Omega(0)\|^2 < \frac{2}{\lambda_{\max}(I)} k_R (1 - \Psi(R(0), R_c(0))) \quad (102)$$

עבור מצב בו שגיאת תנאי ההתחלה גדולה מדי נוכל להשתמש ראשית בבקר האוריינטציה עם $R_d = R_c$ ולהגיע למצב בו מתקיים $\Psi(R, R_c) < 1$ וממצב זה נעבור למצב בקרת מיקום.

דיאגרמת הבלוקים עבור מצב טיסה זה תהיה:



איור 4.3: דיאגרמת הבלוקים עבור מצב בקרת מיקום.

4.2.2 הוכחת יציבות עבור מצב עקיבת מיקום

כעת נוכיח כי $(\bar{e}_x, \bar{e}_y, \bar{e}_\Omega, \bar{e}_R) = (\bar{0}, \bar{0}, \bar{0}, \bar{0})$ היא נקודת שיווי משקל יציבה אקספוננציאלית עבור תנאי ההתחלה שצוינו בסעיף הקודם. בכדי לעשות זאת נמצא תחילה את הדינמיקה של פונקציות השגיאה, לאחר מכן נראה כי שגיאת המהירות חסומה, ובסופו של דבר באמצעות פונקציית ליאפונוב נוכיח יציבות אקספוננציאלית של נקודת שיווי המשקל.

התנאים שהגדרנו ליציבות עבור מצב טיסה זה במשוואות (101) ו- (102) הם למעשה אותם התנאים שהגדרנו עבור מצב הטיסה הקודם (50) ו- (51) רק מחמירים יותר. כלומר, באותה הדרך שהגענו לאי-השוויונים (77) ו- (78) נוכל לקבל כי:

$$k_R \Psi(R, R_c) \leq v_1'(t) \leq v_1'(0) < k_R \quad (103)$$

$$\Psi(R(t), R_c(t)) \leq \gamma_2 < 1 \quad (104)$$

כלומר, תחת האילוצים שהצבנו, פונקציית השגיאה $\Psi(R(t), R_c(t))$ שוכנת בתת-הקבוצה L_{γ_2} , אשר שייכת לקבוצה L_1 , אשר שייכת לקבוצה L_2 ($L_{\gamma_2} \subset L_1 \subset L_2$), כפי שהגדרנו ב- (40), ולכן מתקיים $\Psi(R, R_c) > 0$ סביב $R = R_c$ ובנוסף שגיאת האוריינטציה (ההבדל בין R ל- R_c) קטנה מ- 90° .

נגזור כעת את הביטוי עבור שגיאת המהירות (92) ונכפיל ב- m (מסת הכלי) ונקבל:

$$m\dot{e}_v = m\ddot{x} - m\ddot{x}_d \quad (105)$$

בהצבת הביטוי עבור $m\ddot{x}$ ממשוואות הדינמיקה (31) נקבל כי:

$$m\dot{e}_v = -mg[0 \ 0 \ 1]^T + F_{tot} \cdot R[0 \ 0 \ 1]^T - m\ddot{x}_d \quad (106)$$

אם נתבונן כעת בביטוי $e_3^T R_c^T R \cdot e_3$ (כאשר e_3 ווקטור הבסיס במערכת האינרציאלית המצביע כלפי מעלה - בכיוון z_e) נוכל להגיד כי הוא מייצג את הקוסינוס של הזווית שבין ציר z במערכת האינרציאלית (z_e) לבין ציר z הרצוי (z_{b_c}) במערכת הצמודה לגוף. הסיבה לכך היא שאנו מסובבים תחילה את וקטור הבסיס e_3 לפי R , לאחר מכן מסובבים אותו חזרה לפי R_c ומכפילים אותו בסופו של דבר בווקטור הבסיס המקורי e_3 (למעשה מכפלה סקלארית בין ווקטורים בגודל יחידה). כפי שראינו ב- (80), ניתן לבטא את הקוסינוס של מטריצת הסיבוב היחסית $R^T R_c$ באמצעות הביטוי $\cos(\|\omega\|) = 1 - \Psi(R(t), R_c(t))$. כעת, אם ניזכר בהגבלות שהתקבלו במשוואה (104) עבור $\Psi(R(t), R_c(t))$ נוכל לרשום כי:

$$1 > e_3^T R_c^T R \cdot e_3 > 1 - \Psi(R(t), R_c(t)) > 0 \quad (107)$$

כלומר, הערך $\frac{1}{e_3^T R_c^T R \cdot e_3}$ מוגדר היטב. כעת, נוסיף ונחסיר את הביטוי $\frac{F_{tot}}{e_3^T R_c^T R \cdot e_3} R_c e_3$ לצד הימני של (106) ונקבל כי:

$$m\dot{e}_v = -mge_3 - m\ddot{x}_d + \frac{F_{tot}}{e_3^T R_c^T R e_3} R_c e_3 + X \quad (108)$$

כאשר X נתון ע"י:

$$X = \frac{F_{tot}}{e_3^T R_c^T R e_3} \left((e_3^T R_c^T R e_3) R e_3 - R_c e_3 \right) \quad (109)$$

נגדיר עתה את הווקטור A להיות :

$$A = -k_x e_{\bar{x}} - k_v e_v + m g e_3 + m \ddot{x}_d \quad (110)$$

כעת, ע"י שימוש ב- (93) נוכל לכתוב כי $F_{tot} = A \cdot R \cdot e_3$ וע"י שימוש במשוואה (94) נוכל לכתוב כי $z_{b_c} = R_c e_3 = A / \|A\|$. באמצעות שילוב המשוואות הללו נוכל לקבל כי $F_{tot} = \|A\| \cdot (R_c e_3) \cdot Re_3$. בהצבת כל אלה, עבור האיבר השלישי באגף ימין של המשוואה (108) נקבל כי :

$$\frac{F_{tot}}{e_3^T R_c^T Re_3} R_c e_3 = A \quad (111)$$

ובהצבת הביטוי עבור A לתוך (108) נקבל כי מתקיים :

$$m \dot{e}_v = -k_x e_{\bar{x}} - k_v e_v + X \quad (112)$$

נגדיר עתה את פונקציית ליאפונוב עבור שגיאת המיקום להיות :

$$v_2 = \frac{1}{2} k_x \|e_{\bar{x}}\|^2 + \frac{1}{2} m \|e_v\|^2 + c_2 e_{\bar{x}} \cdot e_v \quad (113)$$

כעת נגזור את פונקציית ליאפונוב הזו לפי הזמן, ונקבל :

$$\dot{v}_2 = k_x e_x \cdot e_v + m e_v \cdot \dot{e}_v + c_2 \|e_v\|^2 + c_2 e_x \cdot \dot{e}_v \quad (114)$$

נציב את הביטוי שהתקבל עבור \dot{e}_v ממשוואה (112) לתוך המשוואה הנ"ל ונקבל :

$$\dot{v}_2 = -(k_v - c_2) \|e_v\|^2 - \frac{k_x c_2}{m} \|e_x\|^2 - \frac{c_2 k_v}{m} e_x \cdot e_v + X \cdot \left(\frac{c_2}{m} e_x + e_v \right) \quad (115)$$

נרצה כעת לקבל חסם עבור X . נעשה זאת באמצעות הצבה של הביטוי $F_{tot} = \|A\| \cdot (R_c e_3) \cdot R \cdot e_3$ לתוך הביטוי עבור X (109), ונקבל :

$$X = \|A\| \cdot \left[R_c e_3 + \left(e_3^T R_c^T R \cdot e_3 \right) \cdot R \cdot e_3 \right] \quad (116)$$

ובהצבת הביטוי עבור X (109), ותחת ההנחה שהנחנו כי $\|m g e_3 + m \ddot{x}_d\| < B$, נוכל לקבל חסם ל- X :

$$\|X\| \leq (k_x \|e_{\bar{x}}\| + k_v \|e_v\| + B) \cdot \left\| \left(e_3^T R_c^T Re_3 \right) Re_3 - R_c e_3 \right\| \quad (117)$$

אם נתבונן באיבר האחרון נוכל להגיד כי הוא מייצג את הסינוס של הזווית שבין $R_c \cdot e_3$ לבין $R \cdot e_3$ וזאת מפני שמתקיים :

$$(e_3 R_c \cdot Re_3) Re_3 - R_c e_3 = Re_3 \times (Re_3 \times R_c e_3) \quad (118)$$

כאשר החלק הימני של המשוואה, משמעותו סינוס הזווית כפול גודל הווקטורים (ווקטורי יחידה במקרה הזה). כמו כן, אם ניוזכר במשוואה (81) נוכל להגיד כי $\|e_R\|$ מייצג את סינוס הזווית של מטריצת הסיבוב היחסית $R^T R_c$. כעת נוכל לכתוב כי :

$$\|X\| \leq (k_x \|e_{\bar{x}}\| + k_v \|e_v\| + B) \cdot \|e_R\| \quad (119)$$

כעת, באמצעות (83) ו- (104) נוכל לקבל כי :

$$\|e_R\| = \sqrt{\Psi(2 - \Psi)} \leq \sqrt{\gamma_2(2 - \gamma_2)} \equiv \xi \quad (120)$$

בהצבת הביטוי לחסם שקיבלנו עבור X חזרה לנגזרת של פונקציית ליאפונוב (115), נקבל כי :

$$\dot{v}_2 \leq -(k_v(1 - \xi) - c_2) \|e_v\|^2 - \frac{c_2 k_x}{m} (1 - \xi) \|e_x\|^2 + \dots \quad (121)$$

$$\dots + \frac{c_2 k_v}{m} (1 + \xi) \|e_{\bar{x}}\| \cdot \|e_v\| + \|e_R\| \left\{ k_x \|e_x\| \|e_v\| + \frac{c_2}{m} B \|e_x\| + B \|e_v\| \right\}$$

בביטוי שקיבלנו עבור הנגזרת של פונקציית ליאפונוב ניתן לראות כי ישנו איבר משולש, כלומר מכפלה של שלושה שגיאות שונות $\|e_R\| \|e_{\bar{x}}\| \|e_v\|$. בכדי להתמודד עם איבר זה נחפש חסם לאחד האיברים, לדוגמא $\|e_v\|$ וכך נקבל תלות בשני שגיאות בלבד. נגדיר עתה $v_2' = v_2|_{c_2=0, k_x=0}$. מהביטויים שקיבלנו במשוואות (113) ו- (121) נקבל כי :

$$\begin{aligned} v_2' &= \frac{1}{2} m \|e_v\|^2 \\ \dot{v}_2' &\leq -k_v (1 - \xi) \|e_v\|^2 + B \|e_v\| \end{aligned} \quad (122)$$

ברור כי כאשר $\|e_v\| > \frac{B}{k_v(1 - \xi)}$ הנגזרת של פונקציית ליאפונוב תהיה שלילית, בעוד הפונקציה עצמה מוגדרת חיובית, כלומר $\|e_v\|$ תקטן. מצד שני, במידה וזהו לא המצב, הערך של $\|e_v\|$ ייגדל (כי הנגזרת חיובית) עד למצב בו התנאי $\|e_v\| > \frac{B}{k_v(1 - \xi)}$ מתקיים. כלומר, הערך $\|e_v\|$ יהיה חסום ע"י :

$$\|e_v(t)\| \leq \max \left\{ \|e_v(0)\|, \frac{B}{k_v(1-\xi)} \right\} \equiv e_{v\max} \quad (123)$$

נתבונן עתה בפונקציית ליאפונוב כללית עבור כל המערכת. נעשה זאת ע"י חיבור של (65) ו- (113) :

$$\dot{v} = \dot{v}_1 + \dot{v}_2 \quad (125)$$

עתה, ע"י שימוש במשוואה (104), בדרך דומה לדרך שהראנו עבור מצב הטיסה הקודם, נוכל להראות כי פונקציית ליאפונוב תהיה חסומה ע"י :

$$z_1^T M_{11} z_1 + z_2^T M_{21} z_2 \leq v \leq z_1^T M_{12}' z_1 + z_2^T M_{22} z_2 \quad (126)$$

כאשר $z_1^T = [\|e_x\| \quad \|e_v\|]$ ו- $z_2^T = [\|e_R\| \quad \|e_\Omega\|]$, והמטריצות מוגדרות בצורה הבאה :

$$M_{11} = \frac{1}{2} \begin{bmatrix} k_R & -c_1 \\ -c_1 & \lambda_{\min}(I) \end{bmatrix}, M_{12}' = \frac{1}{2} \begin{bmatrix} \frac{2k_R}{2-\gamma_2} & c_1 \\ c_1 & \lambda_{\max}(I) \end{bmatrix} \quad (127)$$

$$M_{21} = \frac{1}{2} \begin{bmatrix} k_x & -c_2 \\ -c_2 & m \end{bmatrix}, M_{22} = \frac{1}{2} \begin{bmatrix} k_x & c_2 \\ c_2 & m \end{bmatrix}$$

כעת, ע"י שימוש במשוואות (71), (121) ו- (123) נוכל לקבל :

$$\dot{v} \leq -z_1^T W_1 z_1 + z_2^T W_{12} z_1 - z_2^T W_2 z_2 \quad (128)$$

כאשר המטריצות מוגדרות בצורה הבאה :

$$W_1 = \begin{bmatrix} \frac{k_R c_1}{\lambda_{\max}(I)} & -\frac{k_\Omega c_1}{2\lambda_{\min}(I)} \\ -\frac{k_\Omega c_1}{2\lambda_{\min}(I)} & k_\Omega - c_1 \end{bmatrix}, W_2 = \begin{bmatrix} \frac{k_x c_2}{m} & -\frac{k_v c_2}{2m} (1+\xi) \\ -\frac{k_v c_2}{2m} (1+\xi) & k_v (1-\xi) - c_2 \end{bmatrix} \quad (129)$$

$$W_{12} = \begin{bmatrix} k_x e_{v\max} + \frac{c_2}{m} B & 0 \\ B & 0 \end{bmatrix}$$

ע"י בחירה נכונה של הקבועים c_1, c_2 נוכל להבטיח כי כל המטריצות שהגדרנו ב- (127) ו- (129),

למעט W_{12} יהיו חיובית. נבחר את הקבועים הללו בצורה הבאה :

$$c_1 < \min \left\{ k_\Omega, \frac{4k_\Omega k_R \lambda_{\min}(I)^2}{k_\Omega^2 \lambda_{\max}(I) + 4k_R \lambda_{\min}(I)^2}, \sqrt{k_R \lambda_{\min}(I)} \right\} \quad (130)$$

$$c_2 < \min \left\{ k_v(1-\xi), \frac{4mk_x k_v(1-\xi)}{k_v^2(1+\xi)^2 + 4mk_x}, \sqrt{k_x m} \right\} \quad (131)$$

ובנוסף, נבחר את הקבועים הללו כך שיתקיים :

$$\lambda_{\min}(W_1) > \frac{4\|W_{12}\|^2}{\lambda_{\min}(W_1)} \quad (132)$$

עתה נוכל לרשום כי :

$$\begin{aligned} \lambda_{\min}(M_{11})\|z_1\|^2 + \lambda_{\min}(M_{21})\|z_2\|^2 \leq \nu \leq \lambda_{\max}(M_{12})\|z_1\|^2 + \lambda_{\max}(M_{22})\|z_2\|^2 \\ \dot{\nu} \leq -\lambda_{\min}(W_1)\|z_1\|^2 + \|W_{12}\|_2\|z_1\|\|z_2\| - \lambda_{\min}(W_2)\|z_2\|^2 \end{aligned} \quad (133)$$

ע"י בחירת הקבועים c_1, c_2 לפי המשוואות (130), (131), (132) נוכל להבטיח כי אכן פונקציית ליאפונוב עצמה תהיה מוגדרת חיובית וכי נגזרתה תהיה מוגדרת שלילית. ולכן נקבל כי נקודת שיווי המשקל $(\bar{e}_x, \bar{e}_v, \bar{e}_\Omega, \bar{e}_R) = (\bar{0}, \bar{0}, \bar{0}, \bar{0})$ היא נקודת שיווי משקל אקספוננציאלית.

כפי שכבר ציינו, ברור כי במקרה שתנאי ההתחלה גדולים מהתנאים שצוינו עבור מצב טיסה זה (101), (102), נוכל להשתמש ראשית במצב הטיסה של עקיבת אוריינטציה (שגם הוא משיג עקיבה אקספוננציאלית), עד שנקבל שגיאה קטנה מספיק בכדי להשתמש בבקר עבור מצב טיסה זה.

4.3 מצב בקרת מהירות

מצב טיסה זה דומה מאוד למצב הטיסה הקודם שהוצג (בקרת מיקום). ההבדל העיקרי הוא שבמצב טיסה זה אנו רוצים לעקוב אחרי מהירות רצויה ולא דווקא אחרי מיקום רצוי, כלומר הכניסה לבקר היא v_d .

כניסות הכוח והמומנטים במצב טיסה זה מוגדרים באותה צורה כמו במצב בקרת מיקום, וההבדל הוא שבמצב טיסה זה אנו מתעלמים משגיאת המיקום (אין כזו כי אין מיקום רצוי), ולכן למעשה נשתמש במצב הטיסה הקודם, כאשר במשוואות (93) ו- (94) נבחר את ההגבר k_x להיות אפס.

חשוב להבין כי עקיבה אחר מצב רנדומלי כלשהו ומהירות רנדומלית כלשהי איננה אפשרית מלבד במקרים בודדים, מפני שאמנם כלי הטיס עשוי להגיע למיקום הרצוי, אך בגלל העקיבה אחר המהירות הוא ימשיך להתקדם ויתרחק מן המיקום הרצוי. ברור שעבור מצבים בהם ישנה התאמה בין המיקום הרצוי לבין המהירות הרצויה (כאשר המהירות הרצויה היא נגזרת המיקום הרצוי) ניתן לעקוב אחר מיקום ומהירות יחד, אך במקרה כזה עקיבת מיקום תביא גם לעקיבת מהירות.

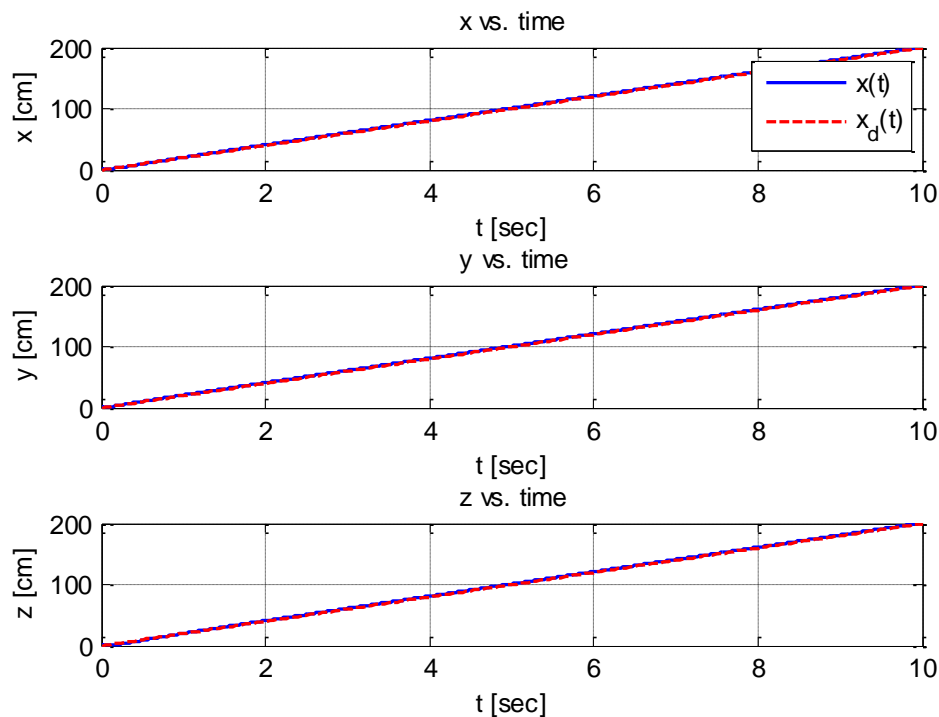
4.4 סימולציות עבור מערכת הבקרה

נרצה לבחון באמצעות סימולציות את התוצאות שמשגיג הבקר. נבצע סימולציות עבור מקרים בהם נרצה כי כלי הטיס יעקוב אחר מיקום משתנה בזמן ונבצע מספר סימולציות עבור מקרים בהם נרצה כי כלי הטיס יעבור ממצב נתון למצב רצוי.

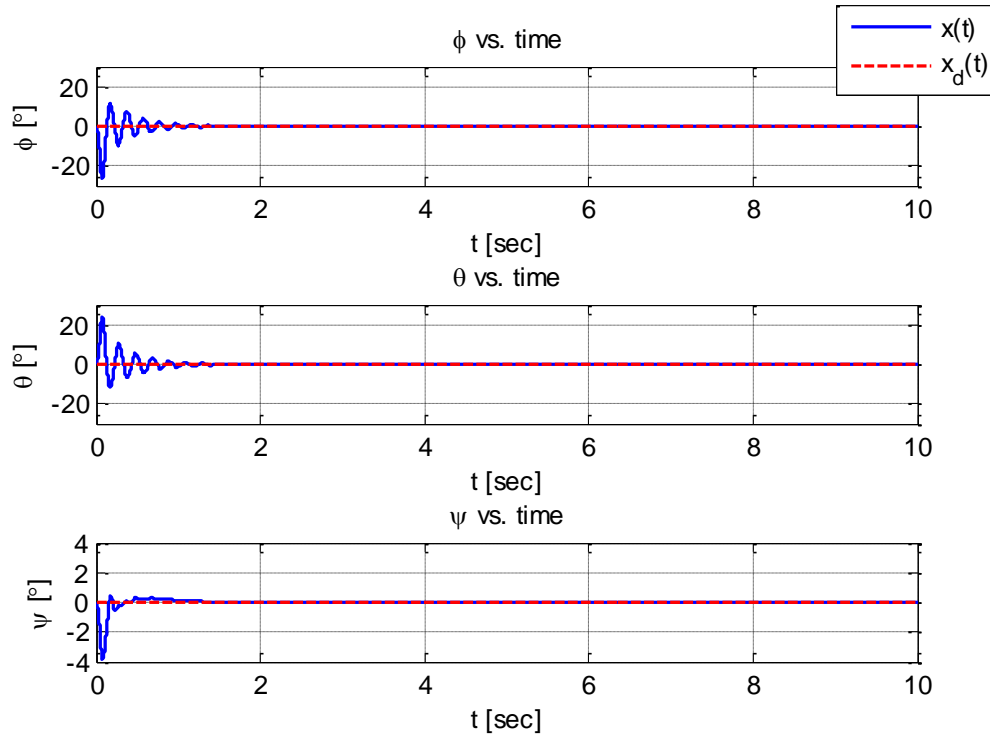
הפרמטרים בהם השתמשנו עבור כלי הטיס הם אילו אשר נמדדו בניסויים עבור כלי הטיס הספציפי בו נשתמש וניתן לראותם בפרק 7. קבועי הבקר בהם בחרנו להשתמש (לאחר בחינת ביצועים) הינם $k_x = 16$, $k_v = 10$, $k_R = 9$ ו- $k_\Omega = 2.5$.

4.4.1 סימולציות עקיבה

עבור סימולציית העקיבה הראשונה נדמה מצב בו כלי הטיס מתחיל מטיסה אופקית סביבה הנקודה $(0,0,0)$ ועוקב אחר קו תלת-מימדי ישר תוך שמירה על מצב אופקי. תנאי ההתחלה הם כדלהלן: $q = (0,0,0,0,0,0)$ והמצב אחריו נרצה לעקוב הוא $x_d(t) = (0.2t, 0.2t, 0.2t, 0, 0, 0)$. להלן התוצאות שהתקבלו:

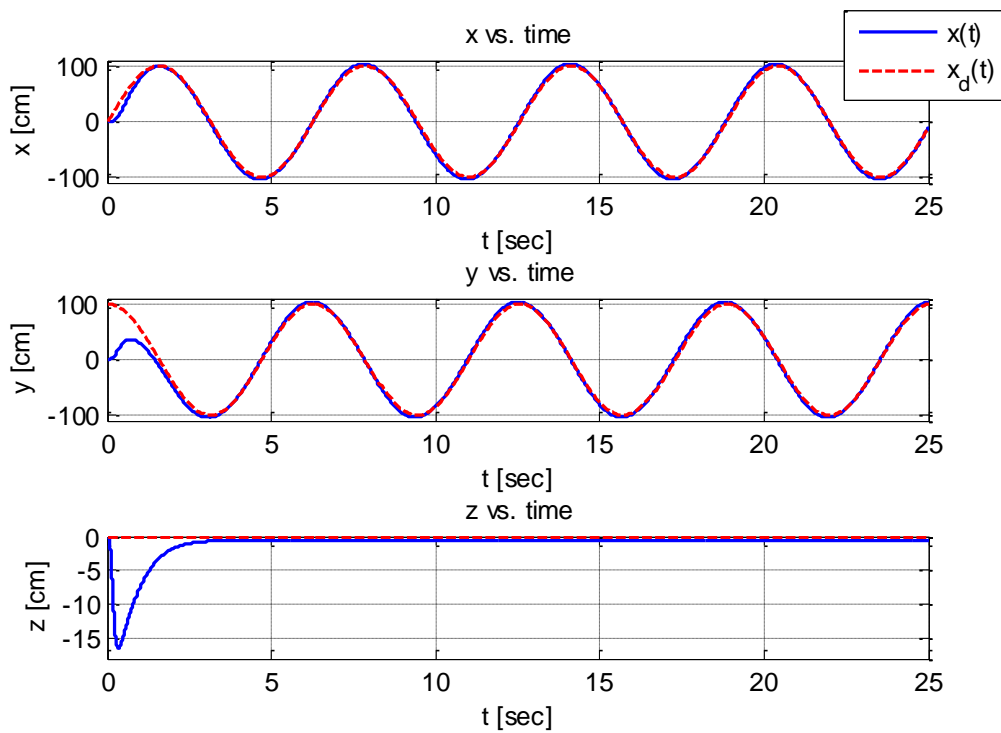


גרף 4.1: מיקום מרכז המסה עבור סימולציית העקיבה הראשונה.

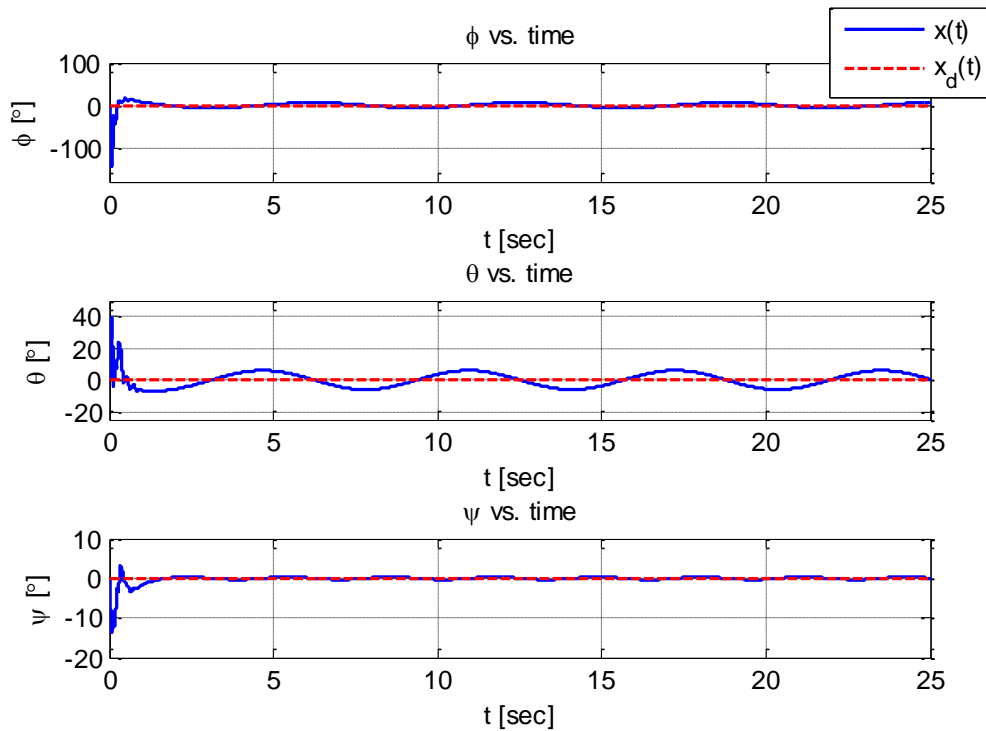


גרף 4.2: זוויות RPY עבור סימולציית העקיבה הראשונה.

עבור סימולציית העקיבה השנייה נדמה מצב בו כלי הטיס מתחיל מטיסה אופקית סביבה הנקודה $(0,0,0)$ ועוקב אחר מעגל אופקי בעל רדיוס של מטר בגובה קבוע תוך שמירה על מצב אופקי. המצב אחרי נרצה לעקוב הוא $x_d(t) = (\sin(t), \cos(t), 0, 0, 0, 0)$. להלן התוצאות:



גרף 4.3: מיקום מרכז המסה עבור סימולציית העקיבה השנייה.



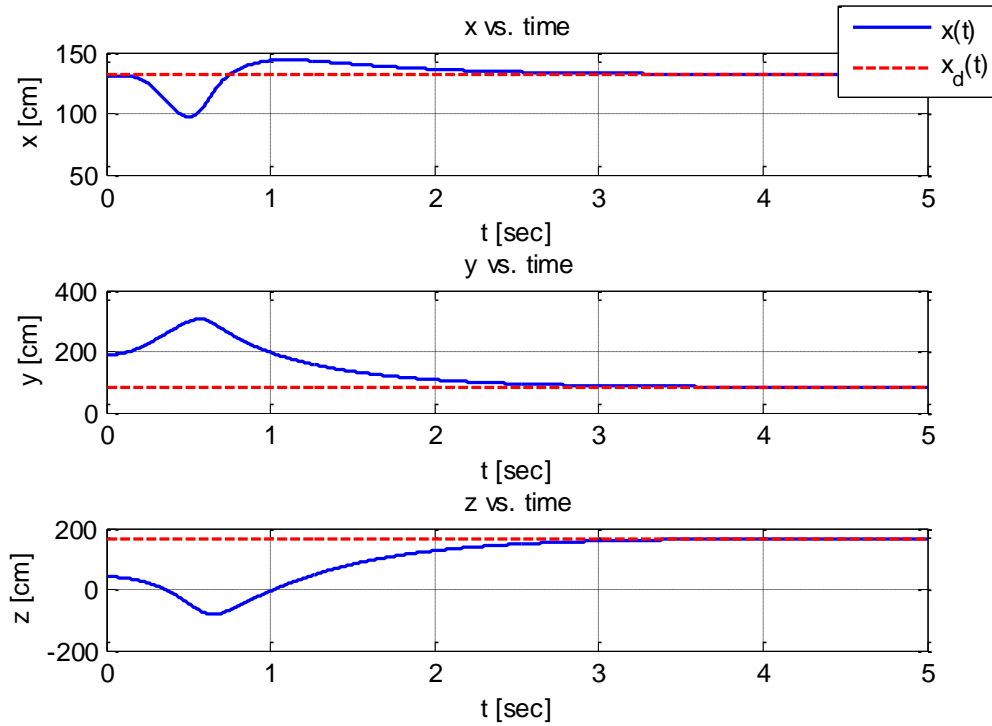
גרף 4.4 : זוויות RPY עבור סימולציית העקיבה השנייה.

4.4.2 סימולציה עבור הגעה ממצב נתון למצב רצוי

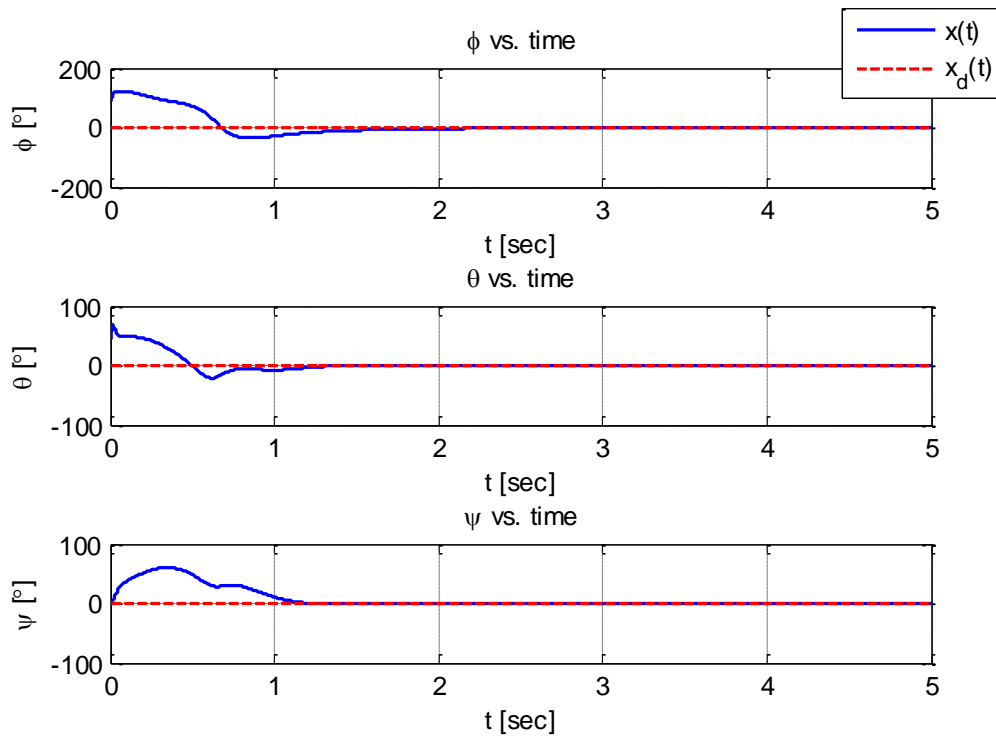
נרצה כעת לבצע סימולציות עבור מצב בו נרצה להביא את כלי הטיס ממצב נתון למצב שיווי משקל רצוי. לשם ביצוע סימולציות אילו ניצור ווקטור מצב רנדומלי אשר ייצג את המצב הנתון ונייצר ווקטור מצב רנדומלי נוסף אשר ייצג את המצב הרצוי.

בהרצת הסימולציה בפעם הראשונה המצב הרנדומלי עבור המוצא הינו $q = (1.3m, 1.9m, 0.42m, 1.5rad, 0.75rad, 0.03rad)$, כאשר $v = (7.5, 3, 2.2)cm/s$ ו- $\Omega = (0.013, 0.015, 0.007)rad/s$ ומיקום מרכז המסה עבור מצב היעד במצב שיווי משקל הינו $q = (1.32m, 0.83m, 1.68m, 0, 0, 0)$. את תוצאות הסימולציה ניתן לראות בגרפים 4.5 ו- 4.6.

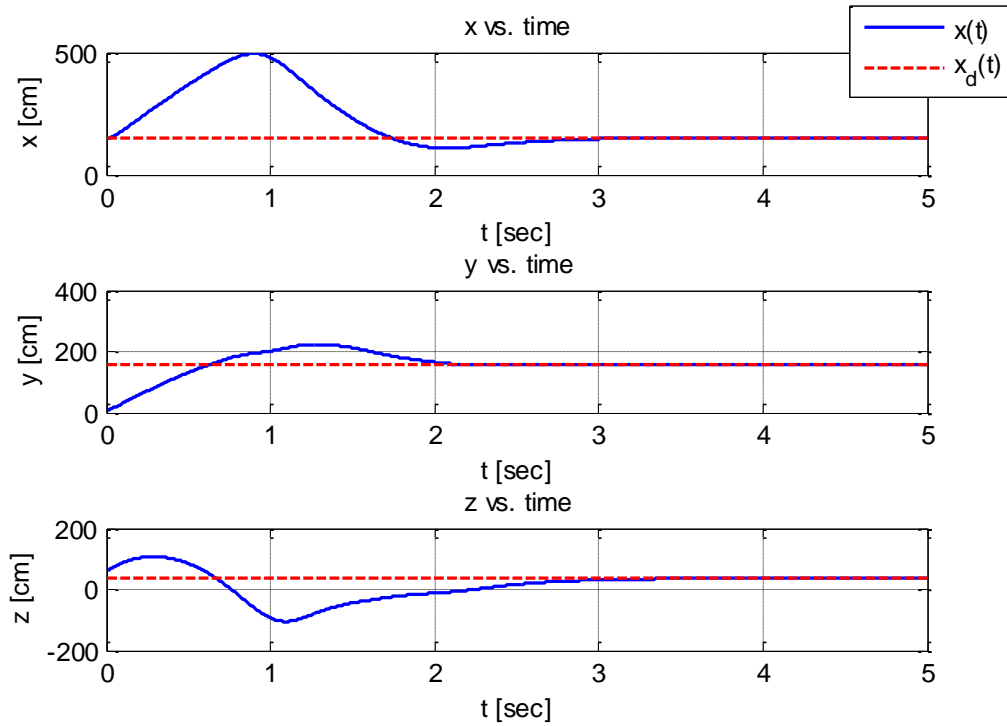
בהרצת סימולציה נוספת המצב הרנדומלי עבור המוצא הינו $q = (1.4m, 0.06m, 0.55m, 0.3rad, 0.6rad, 5.2rad)$, כאשר $v = (3.5, 1.6, 4.8)cm/s$ ו- $\Omega = (0.0007, 0.009, 0.0076)rad/s$ ומיקום מרכז המסה עבור מצב היעד במצב שיווי משקל הינו $q = (1.53m, 1.59m, 0.37m, 0, 0, 0)$. את תוצאות הסימולציה ניתן לראות בגרפים 4.7 ו- 4.8.



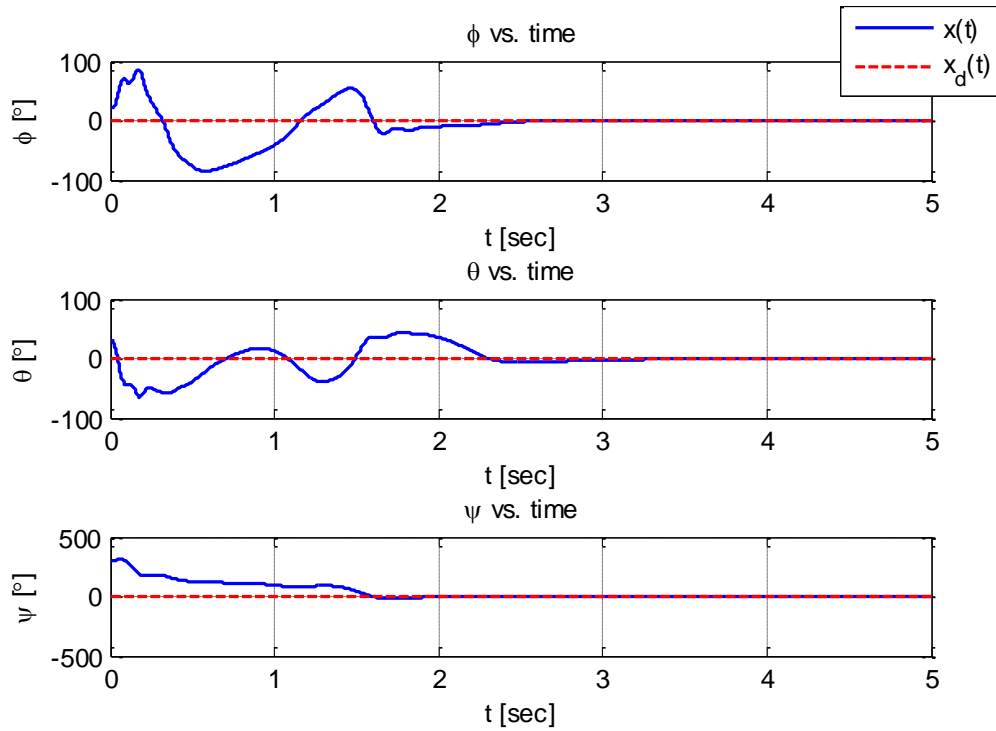
גרף 4.5: מיקום מרכז המסה עבור הסימולציה הראשונה למעבר בין מצבים.



גרף 4.6: זוויות RPY עבור הסימולציה הראשונה למעבר בין מצבים.



גרף 4.7 : מיקום מרכז המסה עבור הסימולציה השנייה למעבר בין מצבים.



גרף 4.8 : זוויות RPY עבור הסימולציה השנייה למעבר בין מצבים.

4.4.3 סיכום תוצאות הסימולציות

מסימולציית העקיבה הראשונה ניתן לראות כי עקיבה אחרי מסלול אשר נקודת ההתחלה שלו זהה לנקודת המצוא של כלי הטיס קרובה למושלמת. עובדה זו הגיונית משום שמערכת הבקרה לא צריכה לפצות על מיקום התחלתי שגוי ולהפעיל כניסות מקסימליות בכדי "להשיג" את האות הרצוי. בנוסף, ניתן לראות כי בזוויות הגלגול והעילרוד (Roll, Pitch) ישנן סטיות קטנות מן המצב הרצוי - ברור כי סטיות אילו נועדות לכיוון של כלי הטיס בכיוון הטיסה הרצוי, וכאמור מדובר בסטיות קטנות.

מסימולציית העקיבה השנייה ניתן לראות כי לכלי הטיס נדרש יותר זמן להתכנס למסלול הרצוי ביחס לסימולציה הראשונה. בסימולציית העקיבה השנייה כלי הטיס היה צריך לפצות על מיקום שגוי מלכתחילה ביחס למסלול הרצוי ולכן ישנן שגיאות גדולות יחסית בזמן ההתחלתי. לאחר מכן ניתן לראות כי האמפליטודה והתדר בפועל כמעט זהים לאילו של המסלול הרצוי. גם במקרה זה ניתן לראות כי ישנן סטיות בזוויות הגלגול והעילרוד - במקרה זה סטיות גדולות יותר, כאשר גם כאן סטיות אילו למעשה נועדו לכיוון את כלי הטיס בכיוון הפעלת הכוח הרצוי שיביא לתנועה הרצויה.

בשתי הסימולציות הראשונות ניתן לראות בנוסף כי זווית הסבסוב (Yaw) כמעט ואיננה משתנה. עובדה זו נובעת מכך שבמצב עקיבת מיקום אנו יכולים לבחור את כיוון היטל ציר ה- x של הגוף, עובדה המתפרשת בפועל כשליטה על זווית הסבסוב של כלי הטיס.

משתי הסימולציות שבוצעו עבור מעבר בין מצבים ניתן לראות כי המיקום והאוריינטציה מתייצבים לאחר ≈ 4 שניות. אמנם מתקבלות שגיאות גדולות בזמן ההתחלתי, אך ההתכנסות היא מהירה. שגיאות אילו הגיוניות ונובעות מהעובדה שהמערכת מתחילה רחוק מן המצב הרצוי וצריכה לפצות על כך במהירות. בנוסף לכלי הטיס ישנה מהירות התחלתית עליה יש להתגבר גם כן.

חשוב לציין כי בחירה שונה של קבועי הבקרה, כמו גם שינוי קטן ביותר של נתוני כלי הטיס יביאו לתוצאות שונות כגון זמן התכנסות, Overshoot וכו'. שינוי קטן של קבוע המהירות k_v לדוגמא יביא להתכנסות מהירה יותר אחר אות משתנה, אך ייגרם לתנודות (אוסצילציות) רבות בזוויות הגלגול והעילרוד של הכלי.

לסיכום, ניתן לראות מן הסימולציות כי הבקר אשר הוצג אכן משיג עקיבה מוצלחת אחר מסלול רצוי, בהתאם לניתוח התיאורטי. בנוסף ראינו כי ניתן להשתמש בבקר למעבר בין מצב טיסה (מיקום ומהירות) מצוי לבין מצב טיסה רצוי בצורה מוצלחת.

5 תכנון מסלול

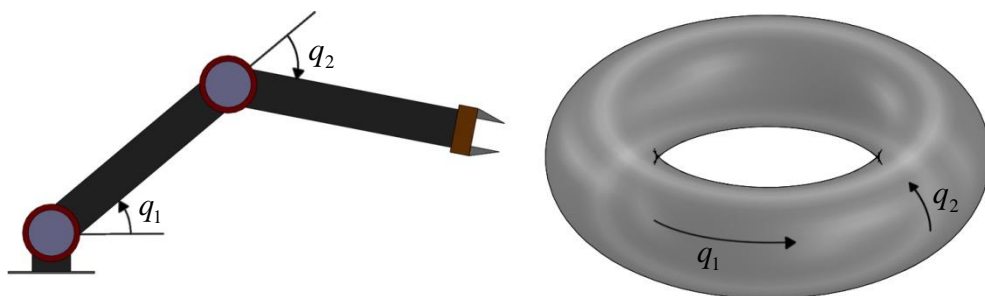
בפרק זה נציג שיטה לתכנון מסלול עבור סביבה מוכרת. בכדי לעשות זאת נסביר תחילה על מרחב הקונפיגורציה. לאחר מכן נציג את המושג "מפת דרכים" (Roadmap) ונציג שיטה לבניית מפת דרכים המבוססת על דיאגרמת וורנוי (Voronoi Diagram), הנקראת גרף וורנוי הכללי (Generalized Voronoi Graph). לסיום נציג את האלגוריתם של Dijkstra למציאת מסלול מנקודה בגרף ליעד. את השיטה המוצגת לתכנון המסלול, כמו גם את האלגוריתם של דייקסטרא, ניתן למצוא במגוון ספרים ומקורות כגון [9] [10].

5.1 מרחב הקונפיגורציה

בכדי לתכנן מסלול (ובשלב מאוחר יותר תנועה) אנו צריכים להיות מסוגלים לציין את המיקום המדויק של כל נקודה ונקודה על הרובוט, וזאת בכדי להימנע מהתקלות במכשולים. כמו כן, מלבד המיקום המדויק יש צורך לדעת היכן נמצאים המכשולים, ואנו צריכים לדעת כיצד להגדירם כך שתהיה להם משמעות. נסביר כעת מהי קונפיגורציה, מהו מרחב הקונפיגורציה, כיצד מגדירים מכשול במרחב הקונפיגורציה ומהי המטרה בזמן תכנון מסלול במושגים של מרחב הקונפיגורציה.

5.1.1 הגדרת מרחב הקונפיגורציה

נגדיר את הקונפיגורציה (Configuration) של רובוט להיות הפירוט המלא של המיקום של כל הנקודות אשר שייכות לגוף הרובוט. כלומר, עבור רובוט בעל n דרגות חופש יש להגדיר n פרמטרים בכדי להגדיר במדויק את המיקום של כל הנקודות השייכות לגוף הרובוט. אם כן, מרחב הקונפיגורציה (Configuration Space) יהיה האוסף של כל הקונפיגורציות האפשריות של הרובוט והוא מסומן באות Q . לפי הגדרה זו, קונפיגורציה מסויימת תהיה נקודה במרחב הקונפיגורציה ותסומן באות q . ע"פ הגדרות אילו, ברור כי המימד של מרחב הקונפיגורציה עבור מערכת מסויימת יהיה לכל הפחות מספר דרגות החופש של אותה המערכת. לדוגמא, ניתן לראות באיור 5.1 את מרחב הקונפיגורציה עבור זרוע מישורית דו-פרקית:



איור 5.1 : מרחב הקונפיגורציה עבור זרוע דו-פרקית.

לזרוע הדו-פרקית המתוארת באיור ישנן שתי דרגות חופש אשר ניתן לבטא באמצעות הזוויות אשר מסומנות באיור (בחלקו השמאלי) ב- q_1, q_2 . כלומר ניתן לבטא את הקונפיגורציה של הגוף באמצעות שני פרמטרים אילו. מרחב הקונפיגורציה יהיה האוסף של כל הקונפיגורציות האפשריות - במקרה הזה ישנו חופש של 360° עבור כל זווית. במקרה הפשוט הנ"ל ניתן לראות את מרחב הקונפיגורציה כטורוס (torus) כפי שניתן לראות בחלקו הימני של האיור, כך שכל נקודה על היקפו תייצג קונפיגורציה מסויימת של הזרוע.

מלבד הגדרת מרחב הקונפיגורציה יש צורך להגדיר גם את המושג מרחב עבודה (Workspace). מרחב העבודה של מערכת נתונה הוא המרחב הגיאומטרי הממשי בו המערכת פועלת והוא מסומן באות W . מרחב העבודה כולל בתוכו מכשולים, כאשר המכשול ה- i מסומן ב- WO_i (מלשון Workspace Obstacle), ובנוסף מרחב העבודה כולל חללים ריקים אותם המערכת יכולה לאכלס. נסמן את המקום המאוכלס ע"י כלל נקודות המערכת במרחב העבודה עבור קונפיגורציה מסויימת q ב- $R_{occ}(q)$.

במקרה שלנו, מרחב העבודה יהיה מרחב אוקלידי תלת-מימדי. כמו כן, למערכת שלנו שש דרגות חופש ולכן מרחב הקונפיגורציה יהיה שישה-מימדי ונוכל להגדיר את משתני הקונפיגורציה כפי שהגדרנו ב- (23).

5.1.2 מכשולים במרחב הקונפיגורציה

לאחר שהגדרנו את המושגים קונפיגורציה ומרחב הקונפיגורציה, נרצה להגדיר בצורה מפורשת מכשולים במרחב הקונפיגורציה. המכשול ה- i במרחב הקונפיגורציה מסומן ב- QO_i ומוגדר כאוסף הקונפיגורציות בהן המערכת מתנגשת עם המכשול WO_i במרחב העבודה, או בניסוח מתמטי:

$$QO_i = \{q \in Q \mid R_{occ}(q) \cap WO_i \neq \emptyset\} \quad (134)$$

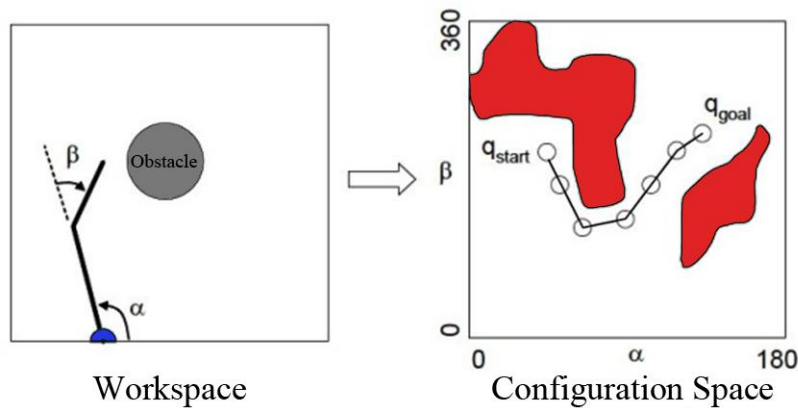
לאחר שנתנו ביטוי למכשולים במרחב הקונפיגורציה, נרצה לתת ביטוי לשאר מרחב הקונפיגורציה, כלומר לחלק של מרחב הקונפיגורציה בו ניתן לנוע בחופשיות. נסמן את החלק החופשי של מרחב הקונפיגורציה ב- Q_{free} , ונגדיר אותו להיות אוסף הקונפיגורציות בהן המערכת לא מתנגשת עם מכשולים כלל, או בניסוי מתמטי:

$$Q_{free} = Q \setminus \left(\bigcup_i QO_i \right) \quad (135)$$

ניתן לראות באיור 5.2 דוגמא לבנייה של מרחב קונפיגורציה, כאשר החלקים הצבועים מייצגים את המכשול, בעוד החלקים הריקים (הלבנים) מייצגים את החלק החופשי של מרחב הקונפיגורציה.

5.1.3 מטרות תכנון המסלול במרחב הקונפיגורציה

נשתמש עתה במושגים שהגדרנו, בכדי להציג את המטרה בעת תכנון מסלול עבור מערכת מסויימת במרחב הקונפיגורציה. המטרה בתכנון מסלול במרחב הקונפיגורציה היא למצוא מסלול כלשהו, אשר לא עובר דרך מכשולים, כך שמקונפיגורציה התחלתית נתונה נוכל להגיע לקונפיגורציה רצויה ע"י עקיבה אחרי המסלול הנ"ל. בניסוח מתמטי, המטרה בתכנון מסלול במרחב הקונפיגורציה היא למצוא העתקה רציפה $c: [0,1] \rightarrow Q_{free}$ כך ψ - $c(0) = q_{start}$ ו- $c(1) = q_{goal}$, כאשר q_{start} היא הקונפיגורציה ההתחלתית ו- q_{goal} היא הקונפיגורציה הרצויה.



איור 5.2 : דוגמא לבניית מרחב קונפיגורציה ומציאת מסלול מתאים.

5.2 מפות דרכים

כעת, לאחר שהגדרנו את המושגים הבסיסיים לתכנון מסלול במרחב הקונפיגורציה, מושגים כמו קונפיגורציה, מרחב הקונפיגורציה ומכשול קונפיגורציה, ולאחר שהבנו מהן המטרות שבתכנון מסלול במושגים אילו, נוכל להציג ולהגדיר את המושג של מפת דרכים (Roadmap). מפת דרכים למעשה ממפה את מרחב הקונפיגורציה החופשי Q_{free} לאוסף של יריעות חד-מימדיות (עקומות) אשר שומרות על הטופולוגיה של מרחב זה. כלומר, מפת דרכים היא למעשה גרף המציג את המאפיינים הטופולוגיים העיקריים של מרחב הקונפיגורציה החופשי.

השימוש במפות דרכים במערכות רובוטיות דומה מאוד לשימוש של אנשים בכבישים מהירים. במקום לתכנן כל מסלול אפשרי הלוקח בחשבון את כל הסמטאות והדרכים הצדדיות, אנו בדו"כ מתכננים את המסלול לרשת של כבישים מהירים, לאחר מכן לאורך הכבישים המהירים ולבסוף מהכבישים המהירים ליעד. בעת תכנון מסלול באמצעות מפת דרכים עקרון הפעולה הוא דומה. ראשית נמצא מסלול חופשי מהמוצא אל מפת הדרכים שהגדרנו, לאחר מכן ננוע לאורך מפת הדרכים לעבר היעד, ולבסוף נמצא מסלול חופשי ממפת הדרכים אל היעד. נגדיר עתה בצורה רשמית את המושג מפת דרכים.

5.2.1 מפת דרכים - הגדרה

מפת דרכים: האיחוד של עקומות חד-מימדיות יהיה מפת דרכים (RM) אם עבור כל מוצא q_{start} ויעד q_{goal} אשר שוכנים ב- Q_{free} וניתנים לחיבור ע"י מסלול, המאפיינים הבאים נכונים:

- נגישות (Accessibility) - קיים מסלול מהמוצא $q_{start} \in Q_{free}$ לנקודה כלשהי על מפת הדרכים $q_{start}' \in RM$.
- עזיבות (Departability) - קיים מסלול מנקודה כלשהי על מפת הדרכים $q_{goal}' \in RM$ לנקודת היעד $q_{goal} \in Q_{free}$.
- קישוריות (Connectivity) - קיים מסלול במפת הדרכים המחבר בין הנקודה $q_{start}' \in RM$ לנקודה $q_{goal}' \in RM$.

כעת, לאחר שהבנו את המושג "מפת דרכים", ע"י בנייה של מפה כזו נוכל למעשה לתכנן את המסלול בין המוצא ליעד ללא התקלות במכשולים (במידה והדבר אפשרי מבחינה מעשית). נשתמש במפת דרכים מהסוג של דפורמציה נסוגה (Deformation Retraction) ע"י שימוש בגרף וורוני הכללי (Generalized Voronoi Graph - GVG).

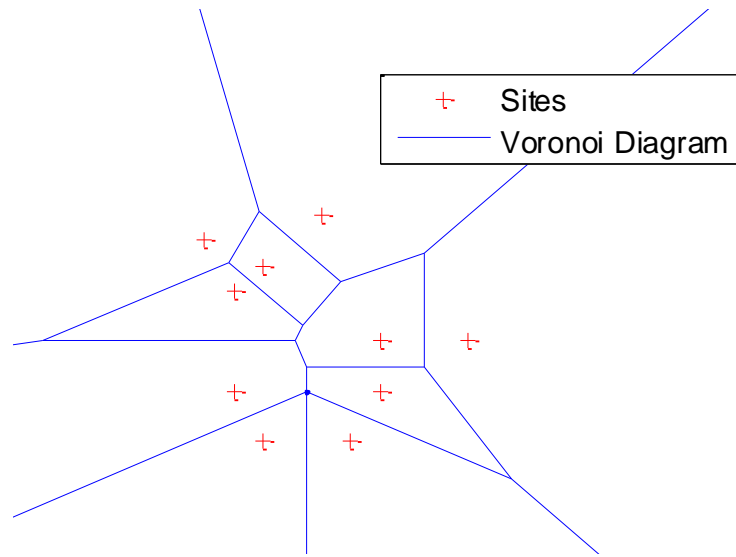
5.3 דפורמציה נסוגה: גרף וורוני הכללי

נסביר תחילה את המושג דיאגרמת וורוני, ולאחר מכן נרחיב מושג זה למושג של דיאגרמת וורוני הכללית (Generalized Voronoi Diagram) ולאחר מכן לגרף וורוני הכללי (GVG). בשלב הבא נסביר מהי דפורמציה נסוגה ונראה כי ה- GVG הוא אכן סוג של דפורמציה נסוגה ועונה על ההגדרות של מפת דרכים. לסיום, נציג את שיטת ה"שריפה בסבך שיחים" (The Brushfire Method), שיטה אחת מיני רבות לבניית גרף וורוני הכללי.

5.3.1 דיאגרמת וורוני

דיאגרמת וורוני מוגדרת עבור קבוצה של נקודות הנקראות "אתרים" (Sites). איזור וורוני מוגדר להיות אוסף הנקודות הקרובות לאתר מסוים. דיאגרמת וורוני מוגדרת כאוסף הנקודות אשר נמצאות במרחקים שווים משני אתרים, כלומר נקודות על הדיאגרמה הן בעלות שני אתרים קרובים. במקרה הדו-מימדי הדיאגרמה היא איחוד של קווים ישרים. ניתן לראות דוגמא לדיאגרמת וורוני עבור אתרים רנדומליים באיור 5.3. לצרכים של תכנון מסלול, אינטואיטיבית ברור כי ע"י הגדרה של המכשולים כאתרים נוכל לקבל את העקומה הרחוקה ביותר בין זוג מכשולים, עבור זוג מכשולים נתון ולכן אוסף העקומות ייתן לנו את מפת הדרכים הבטוחה ביותר. עם הגדרה זו ישנן שתי בעיות - המכשולים במרחב הקונפיגורציה (ובמרחב העבודה) אינם נקודות, ובנוסף המרחב בו מתבצע תכנון התנועה אינו דו-מימדי אלא תלת-מימדי לפחות ברוב המקרים

(עבור קוואדרטור מדובר במרחב קונפיגורציה שישה-מימד). בכדי לפתור בעיות אילו נרחיב את דיאגרמת וורוני לגרף וורוני הכללי.



איור 5.3 : דוגמא לדיאגרמת וורוני עבור אתרים רנדומליים.

5.3.2 גרף וורוני הכללי

נצטרך עתה להרחיב את המושגים עבור אתר, איזור וורוני, הגדרת הדיאגרמה והגדרת גרף וורוני הכללי. נגדיר ראשית את המכשולים להיות האתרים. בנוסף נרצה להגדיר את איזור וורוני הכללי, שהוא הרחבה של איזור וורוני. איזור וורוני הכללי (Generalized Voronoi Region) ה- i מסומן ע"י F_i ומוגדר להיות אוסף הנקודות אשר קרובות ביותר למכשול QO_i , או בניסוח מתמטי נוכל לרשום כי:

$$F_i = \{q \in Q_{free} \mid d_i(q) \leq d_h(q) \forall h \neq i\} \quad (136)$$

כאשר $d_i(q)$ הוא המרחק למכשול QO_i מהנקודה (מהקונפיגורציה) q - $d_h(q)$ הוא המרחק למכשול QO_h מהנקודה q . המרחק בין הנקודה q למכשול QO_i , דהיינו $d_i(q)$, מוגדר להיות המרחק המינימלי בין הנקודה q למכשול QO_i . בצורה מתמטית, $d_i(q)$ מוגדר בצורה הבאה:

$$d_i(q) = \min_{c \in QO_i} d(q, c) \quad (137)$$

כאשר $d(q, c)$ הוא המרחק בין הנקודה c לנקודה q .

דיאגרמת וורוני הכללית מתבססת על נקודות אשר נמצאות במרחקים שווים משני מכשולים QO_i ו- QO_j , כלומר ממשטחים שווי-מרחק (two-equidistant) המוגדרים ע"י:

$$S_{ij} = \{q \in Q_{free} \mid d_i(q) = d_j(q)\} \quad (138)$$

ע"י הגדרת המשטחים שווי המרחק באופן זה נקבל למעשה משטחים אינסופיים. בכדי להימנע מלקבל משטחים אינסופיים, נבנה את דיאגרמת וורונוי הכללית ממשטחים שווי-מרחק סופיים המוגדרים להיות אוסף הנקודות אשר נמצאות במרחקים שווים משני מכשולים QO_i ו- QO_j , כאשר הם זוג המכשולים הקרובים ביותר לנקודות אילו. נוכל להגדיר זאת בצורה הבאה:

$$F_{ij} = \{q \in S_{ij} \mid d_i(q) \leq d_h(q) \forall h\} \quad (139)$$

כעת נוכל להגדיר את דיאגרמת וורונוי הכללית להיות האיחוד של כל המשטחים שווי-המרחק הסופיים:

$$GVD = \bigcup_i \bigcup_j F_{ij} \quad (140)$$

הגדרה זו של דיאגרמת וורונוי הכללית אמנם נכונה למרחב קונפיגורציה מכל מימד, אך כאשר המימד יהיה גדול משניים נקבל משטחים (ואף מרחבים) ולאו דווקא עקומות עליהם נרצה לנוע, ובכדי לקבל עקומות בכל מימד, נרצה להשתמש בגרף וורונוי הכללי (Generalized Voronoi Diagram - GVG).

צורת הבנייה של ה - GVG דומה לזו של ה - GVD. עבור מרחב קונפיגורציה ממימד n גרף וורונוי הכללי מתבסס על נקודות אשר נמצאות במרחקים שווים מ- n מכשולים בדיוק. לדוגמא במקרה התלת-מימדי נגדיר את העקומות בצורה הבאה:

$$S_{ijk} = \{q \in Q_{free} \mid d_i(q) = d_j(q) = d_k(q)\} \quad (141)$$

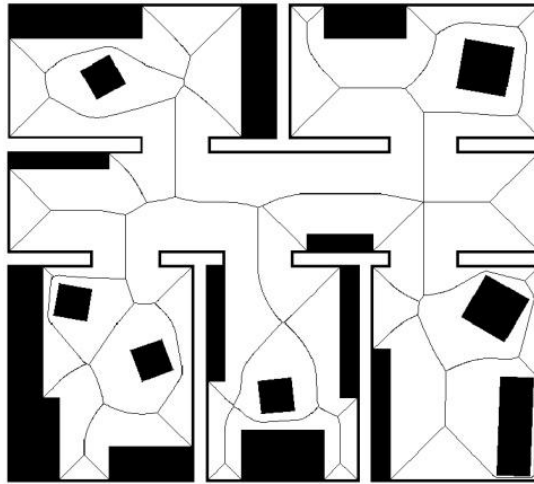
באופן דומה לבניית ה - GVD נדרוש כי המשטחים (העקומות) יהיו סופיים:

$$F_{ijk} = \{q \in S_{ijk} \mid d_i(q) \leq d_h(q) \forall h\} \quad (142)$$

ולסיום נוכל להגדיר את גרף וורונוי הכללי להיות האיחוד של כלל העקומות הללו:

$$GVG = \bigcup_i \bigcup_j \bigcup_k F_{ijk} \quad (143)$$

נציין עתה כי נשתמש במרחב קונפיגורציה תלת-מימדי בלבד מפני שהשימוש במרחב קונפיגורציה ממימד גבוה יותר אינו מעשי - נסביר זאת בהמשך. ניתן לראות דוגמא ל - GVG באיור 5.4.



איור 5.4 : דוגמא לגרף וורנוי הכללי במרחב דו-מימדי.

5.3.3 דפורמציה נסוגה

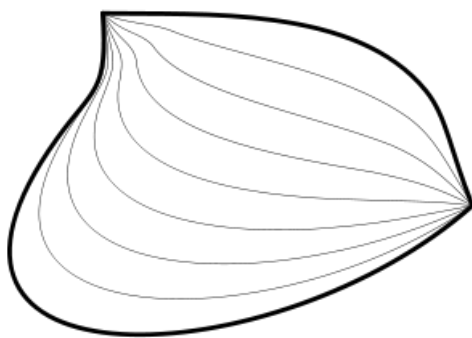
נסביר עתה את המושג דפורמציה נסוגה (Deformation Retract) ונראה כיצד הוא קשור למושג מפת דרכים. לפני שניתן הגדרה פורמלית, ננסה להסביר את המושג של דפורמציה נסוגה באמצעות אנלוגיה. נתבונן בסוכריה בצורת טורוס (איור 5.1) מלא. כשהסוכריה תתמוסס, בסופו של דבר יישאר מעגל. המעגל הזה שומר על הטופולוגיה של המבנה המקורי של הסוכריה. כלומר, אם היינו לוקחים סוכריה ריבועית במקום עגולה, היינו מקבלים בסופו של דבר ריבוע במקום מעגל. המעגל הנ"ל שומר על הטופולוגיה המקורית של הסוכריה וזאת למרות שהוא קטן מאוד ביחס לסוכריה המקורית. באנלוגיה זו הסוכריה מייצגת את מרחב הקונפיגורציה של המערכת, והמעגל שנשאר מייצג את הדפורמציה הנסוגה. הפונקציה שעושה את הפעולה הנ"ל (ממפה טורוס מלא למעגל לדוגמא) נקראת נסיגת דפורמציה (Deformation Retraction).

נגדיר עתה בצורה מתמטית את המושג דפורמציה נסוגה. בכדי להגדיר מושג זה, נגדיר ראשית מושג כללי יותר הנקרא נסיגה (Retract). עבור יריעה X נתונה, נסיגה תהיה פונקציה רציפה $f: X \rightarrow A$ כך ש- A היא תת קבוצה של X , כלומר $A \subset X$, ובנוסף בתחום A הפונקציה היא העתקת הזהות, כלומר $f(a) = a \forall a \in A$.

האוסף של כל הדפורמציות הנסוגות הוא תת-קבוצה של הנסיגות (Retracts). הייחוד של הדפורמציה הנסוגה ביחס לנסיגה הוא שהדפורמציה הנסוגה שומרת על הטופולוגיה של המרחב המקורי. המאפיין החשוב ביותר עליו שומרת הדפורמציה הנסוגה הוא שמספר ה"סוגים" של המסלולים הסגורים במרחב המקורי שווה למספר ה"סוגים" של המסלולים הסגורים לאחר הפעל הפונקציה על המרחב. כלומר, אם נזכר באנלוגיה של הסוכריה, הנסיגה כפי שהגדרנו אותה יכולה להיות מעגל, אך היא יכולה להיות גם נקודה או דיסקה. לעומת זאת, דפורמציה נסוגה יכול להיות מעגל בלבד (בהקשר של הדוגמא הנ"ל).

נגדיר עתה את המושג הומוטופיה (Homotopy). המושג הומוטופיה לקוח מטופולוגיה והגדרתו במילים היא כזו: שתי פונקציות רציפות הינן הומוטופיות ממרחב טופולוגי אחד לשני אם ניתן באופן רציף "לעוות" (to deform) אחת מהן לכדי השנייה. נגדיר עתה הומוטופיה בצורה פורמלית.

נניח כי נתונות שתי פונקציות רציפות $f: U \rightarrow V$ ו- $g: U \rightarrow V$ כאשר U, V הן יריעות. הומוטופיה תהיה פונקציה רציפה $h: U \times [0,1] \rightarrow V$ כך ש- $h(x,0) = f(x)$ ו- $h(x,1) = g(x)$. דוגמא להומוטופיה כזו היא $h(x,t) = (1-t) \cdot f(x) + t \cdot g(x)$. אם קיימת העתקה העונה למאפיינים הללו, אזי f, g הן הומוטופיות. ניתן להגיד כי שני מסלולים הם הומוטופיים אם ניתן לעוות מסלול אחד לתוך המסלול השני. דוגמא להומוטופיה ניתן לראות באיור הבא:



איור 5.5: דוגמא להומוטופיה בין שני מסלולים.

כעת נוכל להגדיר באמצעות שני המושגים הללו (נסיגה והומוטופיה) את המושג דפורמציה נסוגה.

דפורמציה נסוגה - הגדרה:

נניח כי $A \subset X$ ובנוסף נניח כי $f: X \rightarrow A$ היא נסיגה. דפורמציה נסוגה תהיה ההומוטופיה $h: X \times [0,1] \rightarrow X$ כאשר מתקיימים התנאים הבאים:

- $h(x,0) = x$
- $h(x,1) \in A$
- $h(a,t) = a \quad \forall a \in A \quad \forall t \in [0,1]$

כלומר, h (דפורמציה נסוגה) היא הומוטופיה בין נסיגה להעתקת הזהות.

אנו משתמשים בדפורמציה נסוגה בכדי לעוות בצורה חלקה את המרחב X לתת-המרחב A אשר מימדו נמוך יותר (במקרה שלנו חד-מימדי בכדי לקבל גרף). כשנשנה את t בין 0 ל-1, למעשה נזיז נקודה מהמרחב X בצורה רציפה לתת-המרחב A . נוכל להגיד אף יותר מכך, כי נקודה y כלשהי הנמצאת בסביבה של x תזוז בצורה רציפה מ- x ל- A כך שהנקודות החדשות $h(x,t)$ ו- $h(y,t)$ יהיו קרובות האחת לשנייה עבור $t \in [0,1]$.

נרצה עתה להראות כי דפורמציה נסוגה של מרחב הקונפיגורציה אכן עונה על ההגדרה של מפת דרכים, ולאחר מכן נראה כי ה- GVG הוא סוג של דפורמציה נסוגה. התנאי הראשון שנראה כי מתקיים הוא קישוריות (Connectivity). קיים משפט שאומר כי בהינתן שני מרחבים טופולוגיים X ו- Y , ובהינתן הפונקציה הרציפה $f: X \rightarrow Y$, אם קיים מסלול שלם המקשר בין הנקודות

$x_1, x_2 \in X$ אזי התמונות $f(x_1), f(x_2)$ מקושרות גם כן. כלומר, אם קיים מסלול שלם המקשר בין הנקודות במרחב הקונפיגורציה, לאחר הפעלת הדפורמציה הנסוגה (העתקה רציפה) קיים מסלול בין תמונות הנקודות. נסמן $x_1 = q_{start}$, $x_2 = q_{goal}$, $X = Q_{free}$, $f(x_1) = q_{start}$, $f(x_2) = q_{goal}$. שימוש בסימנים אלה בשילוב עם המשפט גורר שאם קיים מסלול בין q_{start} ל- q_{goal} ב- Q_{free} , אזי לאחר הפעלת הדפורמציה הנסוגה ישנו מסלול המקשר בין q_{start} לבין q_{goal} על מפת הדרכים המתקבלת. כלומר הדפורמציה הנסוגה עונה על מאפיין הקישוריות.

בכדי להראות כי גם מאפיין הנגישות (Accessibility) מתקיים, נגדיר $h(x, 0) = q_{start}$ כאשר h היא הדפורמציה הנסוגה. המסלול מהנקודה במרחב הקונפיגורציה למפה יהיה $h(x, \cdot): [0, 1] \rightarrow Q_{free}$ וע"פ ההגדרה $h(x, 1)$ יהיה אלמנט על המפה. לכן, הדפורמציה הנסוגה עונה על מאפיין הקישוריות כך ש- $h(x, 0) = q_{start}$ ו- $h(x, 1) = q_{goal}$. באופן דומה ניתן להראות כי הדפורמציה הנסוגה עונה גם על תנאי העזיבות (Departability). לסיכום, הדפורמציה הנסוגה עונה על שלושת המאפיינים של מפת דרכים, לכן דפורמציה נסוגה היא סוג של מפת דרכים.

לאחר שראינו כי דפורמציה היא סוג של מפת דרכים, נרצה להראות כי גרף וורונוי הכללי הוא סוג של דפורמציה נסוגה ולכן הוא גם מפת דרכים שבאמצעותה ניתן לבצע תכנון מסלול. במהלך בניית גרף וורונוי הכללי אנו למעשה מעוותים בצורה חלקה את המרחב Q_{free} אל מפת הדרכים RM . מעצם ההגדרה של ה- GVG ברור כי נקודות הנמצאות על המפה מועתקות באמצעות העתקת הזהות, כלומר מתקיים $h(q, t) = q \quad \forall q \in RM \quad \forall t \in [0, 1]$. בנוסף, ניתן בצורה חלקה להעתיק נקודה מ- Q_{free} אל RM כך שההגדרה של ה- GVG תענה על שני המאפיינים של דפורמציה נסוגה. דוגמה להעתקה כזו היא:

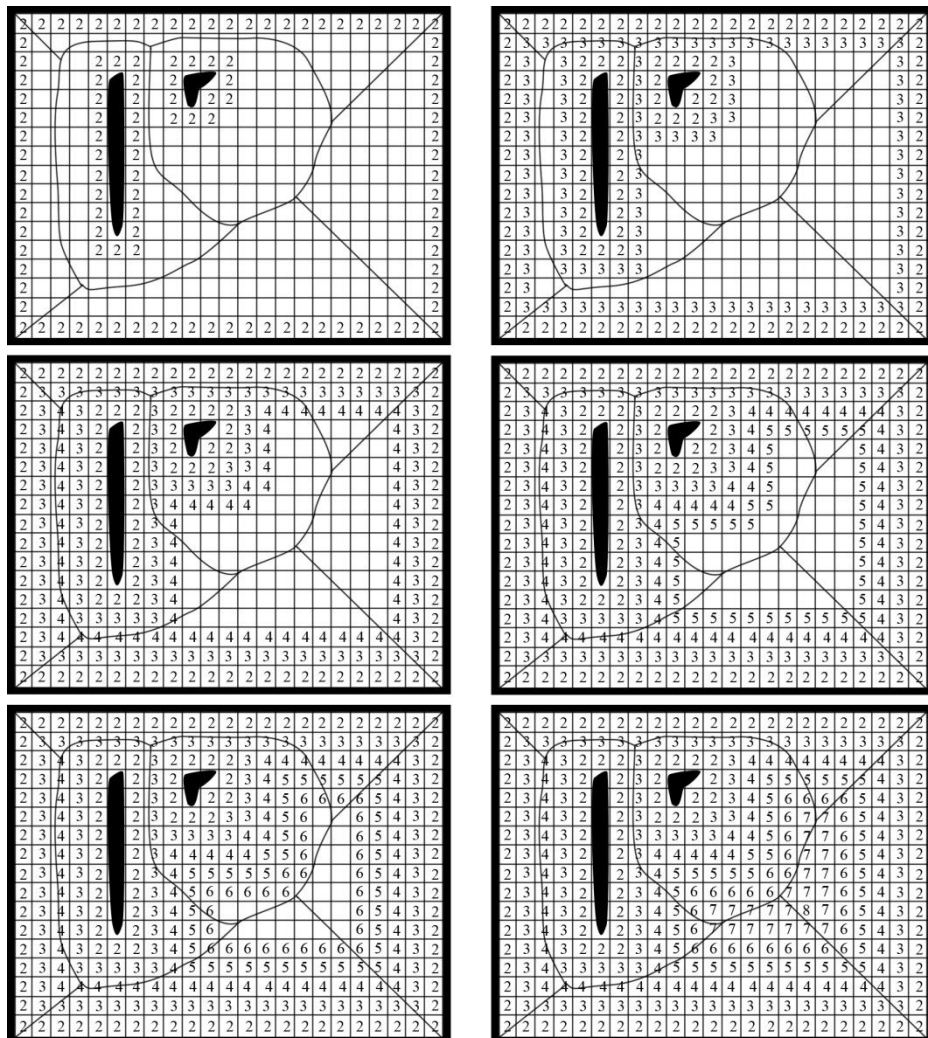
$$\frac{dc(t)}{dt} = \nabla d(c(t)) \quad c(0) = q \in Q_{free} \quad (144)$$

כלומר ע"י התרחקות מהמכשול הקרוב ביותר (התרחקות בכיוון הגרדיאנט) נוכל לקבל העתקה מתאימה. מאחר וה- GVG עונה על המאפיינים של דפורמציה נסוגה, ניתן להגדיר כי ה- GVG הוא מפת דרכים.

5.3.4 אלגוריתם ה- Brushfire

נציג עתה את אלגוריתם ה- Brushfire (התפשטות בשדה קוצים) לבניית ה- GVG . שיטה זו מבוססת על דיסקרטיזציה של מרחב הקונפיגורציה ו"התפשטות" מן המכשולים (כאשר באנלוגיה המכשולים הם מוקדי שריפה) אל עבר המרחב החופשי Q_{free} (כאשר באנלוגיה המרחב החופשי הוא שדה קוצים) עד שהמרחב החופשי מכוסה. שיטת ה- Brushfire היא למעשה שיטה למדידה

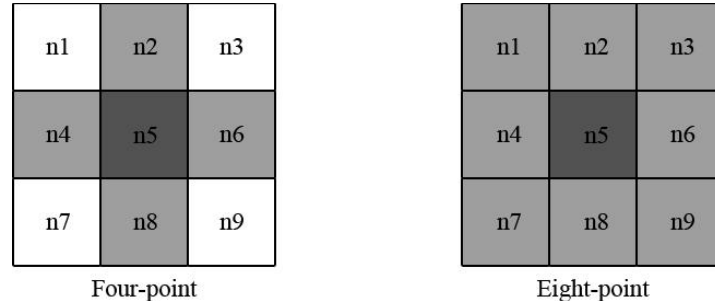
של מרחק על רשת (Grid). ה - Brushfire הוא אלגוריתם אשר מקבל מטריצה (המייצגת את מרחב הקונפיגורציה), כאשר התאים בהם נמצאים מכשולים מאותחלים עם הערך 1 והתאים החופשיים מאותחלים עם הערך 0, ומחזיר מטריצה מלאה במספרים אשר מייצגים את המרחק של כל תא במטריצה מהמכשול הכי קרוב. בשלב הראשון האלגוריתם נותן את הערך 2 לתאים בעלי הערך 0 אשר נמצאים בסמוך לתא עם הערך 1. בשלב הבא, האלגוריתם נותן את הערך 3 לתאים בעלי הערך 0 אשר נמצאים בסמוך לתא עם הערך 2. נמשיך בצורה זו עד אשר "השריפה תשרוף את כל הקוצים" (מכאן נגזר שם האלגוריתם), כלומר עד שלכל התאים במטריצה יהיה ערך השונה מאפס. ככל שהערך בתא מסויים גדול יותר, כך התא הנ"ל למעשה רחוק יותר מהמכשול הקרוב אליו ביותר. נוכל להסתכל על ההתפשטות הנ"ל כעל גלים היוצאים מהמכשולים. ברור כי כאשר שני גלים כאלה נפגשים, המרחק בין שני מכשולים יהיה שווה ולכן הנקודות הללו ישתייכו ל - GVG (במקרה הדו-מימדי. במקרה התלת-מימדי מדובר בהתנגשות של שלושה גלים). ניתן לראות דוגמא לפעולת האלגוריתם באיור הבא:



איור 5.6: דוגמא לבניית ה - GVG באמצעות אלגוריתם ה - Brushfire.

בכדי להשתמש באלגוריתם הנ"ל נותר להגדיר מהי הסביבה הקרובה לתא כלשהו, כלומר עבור תא נתון בעל ערך נתון, אילו תאים סביבו צריכים לעבור שינוי. הגדרה של סביבה קרובה לתא נעשית לרוב באחת משתי הדרכים הבאות: ארבעה תאים סמוכים, או שמונה תאים סמוכים.

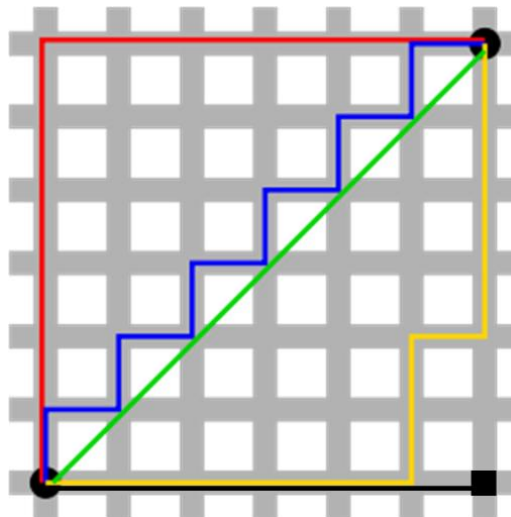
הגדרה באמצעות ארבעה תאים סמוכים משמעותה כי התאים המוגדרים כסביבה הקרובה לתא נתון הם התא מעליו, התא מתחתיו, התא מימינו והתא משמאלו. הגדרה באמצעות שמונה תאים סמוכים משמעותה כי התאים הסמוכים לתא נתון הם כל התאים אשר נוגעים בתא הנתון. האיור הבאה מתאר זאת בצורה ברורה יותר:



איור 5.7: ההגדרות השונות לסביבה קרובה לתא.

שימוש באלגוריתם הנ"ל במרחבים ממימד גבוה משתיים יעשה באותו האופן. עבור מרחב תלת-מימדי המעבר הוא אינטואיטיבי: ארבעה תאים סמוכים הופכים להיות שישה תאים סמוכים (הוספה של התא מלפנים והתא מאחור), שמונה תאים סמוכים הופכים להיות עשרים ושישה תאים סמוכים (כל התאים הנוגעים בתא). במרחבים ממימדים גבוהים יותר נפעל בצורה דומה.

לכל אחת מההגדרות הללו יתרון משלה. הגדרה של סביבה קרובה באמצעות שמונה תאים תבאים לכיסוי מהיר יותר של המרחב, כלומר בניית ה- GVG תהיה מהירה יותר. לעומת זאת, הגדרה באמצעות ארבעה תאים סמוכים תשמור על "מרחק מנהטן" (Manhattan Distance), כלומר על הנורמה L_1 . את היתרון לשימוש במרחק מנהטן ניתן לראות באיור הבא:



איור 5.8: שמירה על מרחק מנהטן ביחס לשימוש בשמונה תאים סמוכים.

כפי שניתן לראות מהאיור, בשימוש בארבעה תאים סמוכים ושמירה על מרחק מנהטן, כל המסלולים בין הנקודות (בצהוב, כחול ואדום) הם בעלי אותו האורך (מרחק). כלומר, אין חשיבות לסדר עידכון התאים באלגוריתם. בנוסף, ע"י שימוש בשמונה תאים סמוכים, הקו הירוק והקו השחור היו מקבלים מרחקים זהים, כאשר ברור כי הקו הירוק ארוך מהשחור.

להלן הפסאודו-קוד (Pseudocode) לבניית גרף וורונוי הכללי באמצעות אלגוריתם ה - Brushfire :

```

1  function Brushfire (Workspace, ObstacleMap):
2      counter = 1 ;
3      GVG = zero array the size of Workspace ;
4      while Workspace includes zeros:
5          for each cell in Workspace that holds value of counter:
6              for each neighbor of cell:
7                  if neighbor value = 0:
8                      neighbor = counter ;
9                      neighbor in ObstacleMap = cell in ObstacleMap ;
10         counter = counter + 1 ;
11     for each cell in ObstacleMap:
12         for each neighbor of cell:
13             if cell  $\neq$  neighbor:
14                 if cell in Workspace  $\neq$  1:
15                     cell in GVG = 1 ;
16     return GVG ;

```

איור 5.9 : פסאודו-קוד לבניית ה - GVG באמצעות אלגוריתם ה - Brushfire.

5.4 האלגוריתם של דייקסטרא

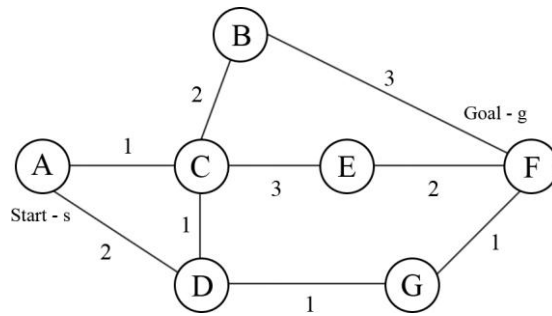
לאחר בניית מפת הדרכים עבור מרחב הקונפיגורציה הנתון וחיבור קונפיגורציות המוצא והיעד אל מפת הדרכים, נרצה למצוא את המסלול הקצר ביותר המחבר בין קונפיגורציית המוצא לקונפיגורציית היעד על מפת הדרכים. לשם כך נשתמש באלגוריתם של דייקסטרא (Dijkstra's Algorithm). נציג את האלגוריתם ונחשב את סיבוכיותו.

5.4.1 פעולת האלגוריתם

השימוש באלגוריתם של דייקסטרא הוא לצורך מציאת המסלול הקצר ביותר מצומת מוצא נתונה המסומנת ב- s לצומת היעד אותה נסמן ב- g (למעשה לכל הצמתים, אך ניתן להפסיק את פעולתו לאחר מציאת המסלול הקצר ביותר ליעד) בגרף. נשתמש בסימון הבא עבור גרף:

$$G = (V, E) \quad (144)$$

כאשר G יהיה הגרף, V כל הצמתים בגרף ו- E הווקטורים (או הקטעים) המחברים בין צומת לצומת. לצורך דוגמא, נתבונן באיור הבא, עבור גרף בעל שבעה צמתים (A, B, C, D, E, F, G) ובעל תשעה ווקטורים (המחברים בין הצמתים):



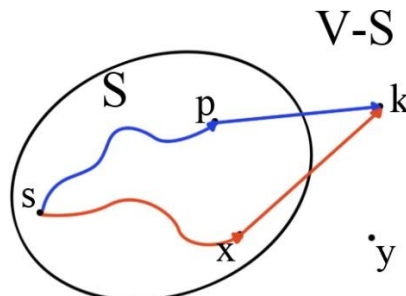
איור 5.10: דוגמא למבנה של גרף בעל שבעה צמתים.

נסמן את המרחק האמיתי בין שני הצמתים k, p ב- $l(k, p)$. לדוגמא, עבור הגרף שבאיור 5.10, $l(C, B) = 2$. במקרה ואין קטע המחבר ישירות בין שני צמתים, נגדיר את המרחק ביניהם להיות אינסוף. $l(C, G) = \infty$, עבור הגרף שבאיור, עבור הגרף שבאיור.

עבור צומת נתון כלשהו בגרף $p \in G$, נגדיר את $d(p)$ להיות המרחק האמיתי של המסלול הקצר ביותר בין המוצא s לבין אותו הצומת p . למעשה, אנו מחפשים את המסלול שייתן לנו את $d(g)$.

נגדיר את הקבוצה S להיות כל הצמתים p מתוך V להן האלגוריתם כבר קבע את המרחק $d(p)$. כאשר הצומת g תהיה שייכת לקבוצה S למעשה מצאנו את המסלול הקצר ביותר. האלגוריתם פועל בצורה אינדוקטיבית, כאשר נתחיל עם המקרה הפשוט ביותר בו רק $s \in S$ ועיי אותן פעולות אשר נבצע עבור המקרה הכי פשוט נמצא את הפתרון.

במצב ההתחלתי $s \in S$ והמרחק $d(s) = 0$, שכן זהו המרחק האמיתי הקצר ביותר בין המוצא לעצמו. לאחר מכן, עבור כל צומת $k \in V - S$ (כלומר שעדיין לא נקבע להם המרחק האמיתי הקצר ביותר) האלגוריתם ימצא את המרחק הקצר ביותר האפשרי בין s ל- k תוך שימוש בצמתים אשר נמצאים ב- S בלבד, מלבד הצומת k . ניתן לראות זאת באיור הבא:



איור 5.11: המחשת מציאת המרחק הקצר ביותר באמצעות האלגוריתם.

כלומר, עבור כל צומת k ב- V האלגוריתם ייקבע את המרחק הקצר ביותר לצומת בצורה הבאה:

$$d'(k) = \min_{p \in S} [d(p) + l(p, k)] \quad (145)$$

כעת, לאחר קביעת המרחק $d'(k)$ עבור כל $k \in V - S$, נבחר את הצומת בעל הערך הקטן ביותר עבור $d'(k)$ ונוסיף אותה לקבוצה S . כמו כן, נשמור את הווקטור המחבר את הצומת p לצומת k . נמשיך בצורה זו עד שנוסיף את g ל- S .

להלן הפסאודו-קוד למציאת המסלול הקצר ביותר מצומת מוצא נתון לצומת יעד נתון על גרף:

```

1  function Dijkstra (Graph, source, goal):
2       $S = \{s\}$ ;  $\text{dist}[s] = 0$ ;  $\tilde{E} = []$ ;
3       $V =$  the set of all nodes in Graph;
4      while  $V$  is not empty:
5          for each node  $k$  in  $V$ :
6              for each node  $p$  in  $S$ :
7                  find node  $k$  to minimize  $[d(p) + l(p,k)]$ ;
8                  remove  $k$  from  $V$  and add it to  $S$ ;
9                  add edge  $(p,k)$  to  $\tilde{E}$ ;
10                 if  $k = \text{goal}$ :
11                     return  $\tilde{E}$ ,  $\text{dist}$ ;
12     return  $\tilde{E}$ ,  $\text{dist}$ ;

```

איור 5.12: פסאודו-קוד למציאת המסלול הקצר ביותר באמצעות האלגוריתם של דייקסטרה.

בסופו של דבר, הווקטורים אשר נמצאים ב- \tilde{E} יוצרים עץ המחבר את המוצא ליעד (או לכל שאר הצמתים, תלוי בתנאי העצירה שנבחר), כך שכל צומת מחובר באמצעות ווקטור אחד לעץ. עץ זה נקרא "Dijkstra's Shortest Path Tree".

חשוב לציין כי במידה והמרחק הקצר ביותר המתקבל בין המוצא ליעד הוא אינסוף, משמעות הדבר היא כי לא קיים מסלול אשר מחבר את המוצא ליעד. כלומר האלגוריתם שלם - במידה וקיים פתרון האלגוריתם יחזיר את הפתרון הטוב ביותר ובמידה ולא קיים פתרון נקבל תשובה חד-משמעית כי לא קיים פתרון.

5.4.2 סיבוכיות האלגוריתם

לאחר שהסברנו את אופן פעולת האלגוריתם והוכחנו את נכונותו, נרצה לחשב את סיבוכיות האלגוריתם - לתת אומדן לזמן הריצה של האלגוריתם כתלות במספר הצמתים.

כאשר הצומת p ממוקם בתוך הקבוצה S , לכל צומת $k \notin S$ יש לעדכן את הערך $d'(k)$ להיות $d'(k) = \min[d'(k), d(p) + l(p,k)]$. הערך $d'(k)$ מתעדכן פעם אחת בדיוק כתוצאה מהוספה של הצומת p לקבוצה S . לכן, במהלך כל ריצת האלגוריתם, הסיבוכיות עבור עדכוני המרחק הללו תהיה $\sim O(|E|)$.

כעת על האלגוריתם לבחור את הצומת בעל הערך הקטן ביותר של $d'(k)$. ניתן לעשות זאת באמצעות סריקה פשוטה של כל הצמתים ($O(|V|)$), אך דרך יעילה יותר תהיה שמירה על תור מסודר, כך שניתן להוסיף ערכים, לשלוף את הערך המינימלי ולעדכן את הערכים בסיבוכיות של $O(\log|V|)$.

לסיכום, הסיבוכיות של האלגוריתם של דייקסטרא תהיה $O(|E| \cdot \log|V|)$.

5.5 סימולציות לתכנון מסלול

נראה עתה סימולציות שבוצעו עבור בניית מרחב הקונפיגורציה עבור מרחבי עבודה מוגדרים. לאחר מכן נראה את התוצאות המתקבלות מבניית גרף וורונוי הכללי עבור המרחבים הללו. לבסוף נשתמש באלגוריתם של דייקסטרא בכדי למצוא את המסלול בין המוצא ליעד עבור גרפי הוורונוי שהתקבלו עבור מרחבי העבודה המוגדרים.

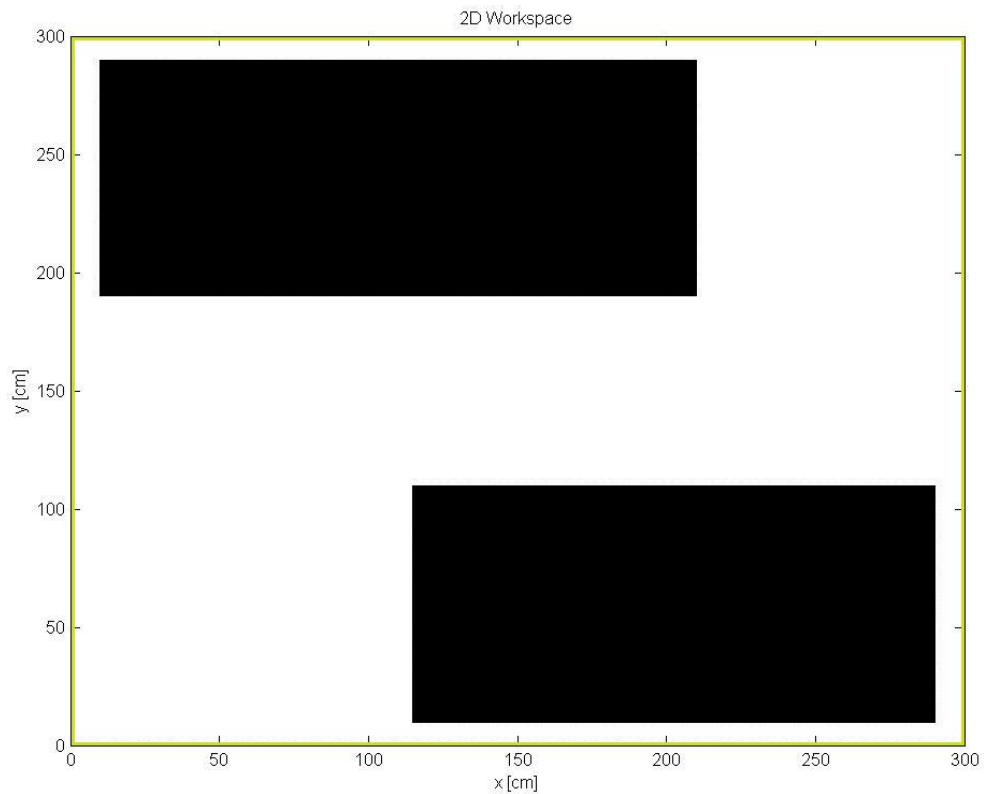
5.5.1 סימולציות לבניית מרחב הקונפיגורציה

גודלו של מרחב העבודה עליו נבצע סימולציה יהיה זהה לזה אשר קיים במעבדה ובו ניתן לבצע ניסויים. גודל מרחב העבודה יהיה $3m \times 3m \times 3m$. נבצע דיסקרטיזציה למרחב העבודה לפי ס"מ - כלומר בשביל להחזיק מטריצה המייצגת את מרחב העבודה יש להשתמש במטריצה בעלת מימדים של $300 \times 300 \times 300$.

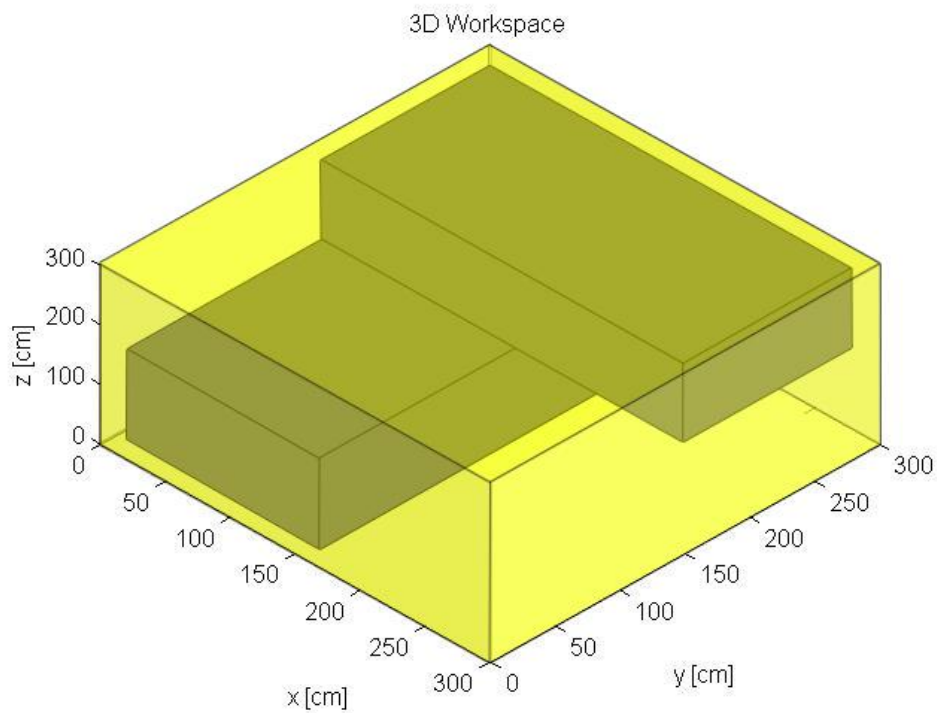
חשוב להבין כי בכדי לבנות את מרחב הקונפיגורציה המלא עבור מרחב עבודה בגודל כזה, נצטרך להשתמש במטריצה שמכילה 300^6 תאים (בכדי לייצג כל מיקום וכל אוריינטציה). ברור כי במחשב סטנדרטי אין די זיכרון בכדי להחזיק מטריצה בגודל כזה - למרות שהיא מכילה אפסים ואחדות בלבד. בנוסף, משך הזמן אשר יידרש לבנייה של מרחב קונפיגורציה מלא שכזה הוא ארוך מדי ולא מעשי כלל. לצורך המחשה בלבד, נבנה מרחבי קונפיגורציה דו/תלת-מימדיים כאשר ההנחה היא כי במצב דו-מימדי גוף הכלי הוא דיסקה ובמצב תלת-מימדי גוף הכלי הוא כדור, כלומר נזניח את אוריינטציית הגוף. בשלב מאוחר יותר (פרק 6) נראה כיצד מתמודדים עם בעיה זו.

מרחב העבודה הדו-מימדי לו נבנה את מרחב הקונפיגורציה יהיה בצורת ריבוע בגודל $300cm \times 300cm$ אשר צלעותיו קיימות ובעל שני מכשולים ריבועיים, עבור דיסקה בקוטר של $60cm$ (ע"פ מידות הכלי שנבנה), ומרחב העבודה התלת-מימדי לו נבנה את מרחב הקונפיגורציה יהיה בצורת קובייה בגודל $300cm \times 300cm \times 300cm$ אשר פאותיה קיימות ובתוכה שני מכשולים תיבתיים (נציין כי השימוש במכשולים תיבתיים נובע מנוחות בלבד, וכי צורת המכשול תשנה במעט את צורת מרחב הקונפיגורציה אך התוצאה הסופית תהיה דומה), עבור כדור בקוטר של $60cm$. ניתן לראות את מרחב העבודה הדו-מימדי באיור 5.13 ואת מרחב העבודה התלת-

מימדי באיור 5.14, כאשר המרחב החופשי הוא לבן, הגבולות צבועים בצהוב והמכשולים צבועים בשחור.

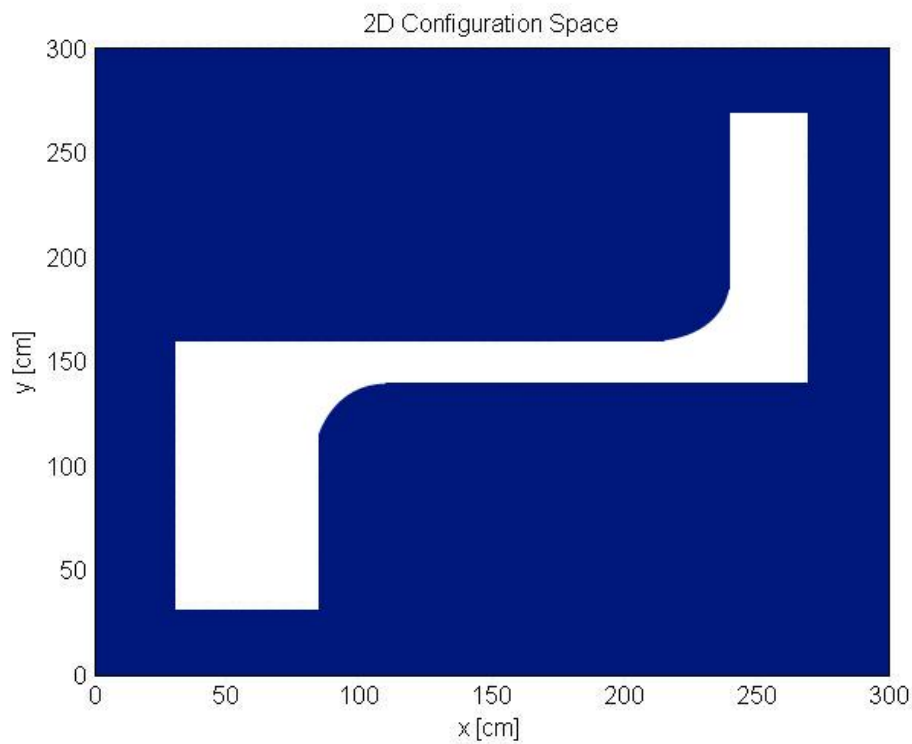


איור 5.13 : מרחב העבודה הדו-מימדי לו נבנה מרחב קונפיגורציה.

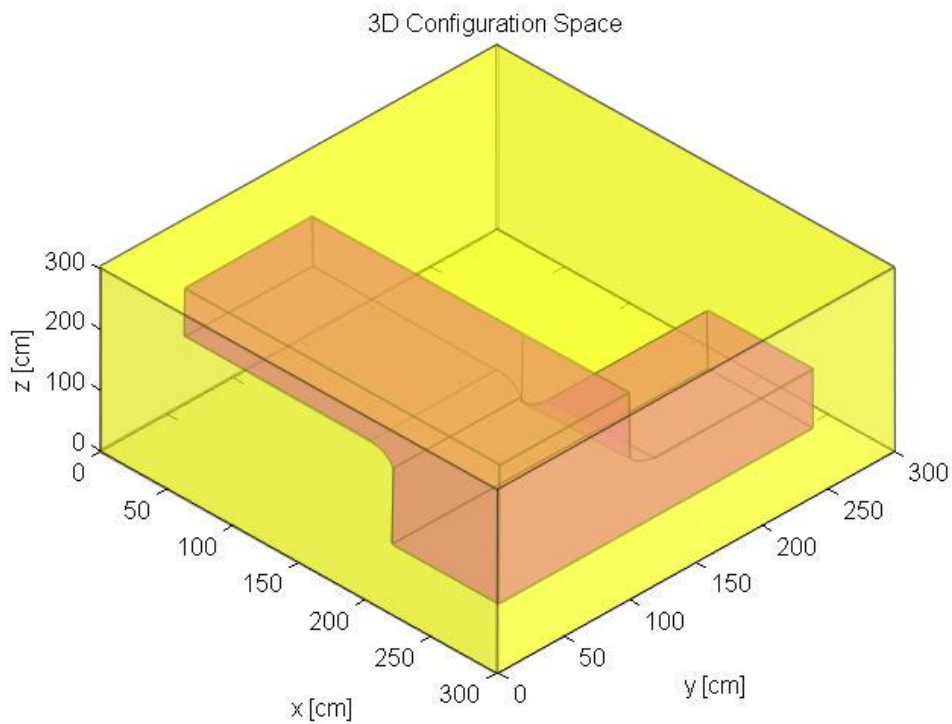


איור 5.14 : מרחב העבודה התלת-מימדי לו נבנה מרחב קונפיגורציה.

לאחר בניית מרחב הקונפיגורציה באמצעות תוכנת ה-MatLab, להלן מרחבי הקונפיגורציה שהתקבלו (כאשר בדו-מימד החלק הלבן הוא המרחב החופשי, ובתלת-מימד הנפח הכולל בין המשטחים הסגולים הוא המרחב החופשי):



איור 5.15 : מרחב הקונפיגורציה שהתקבל עבור מרחב העבודה הדו-מימדי.



איור 5.16 : מרחב הקונפיגורציה שהתקבל עבור מרחב העבודה התלת-מימדי.

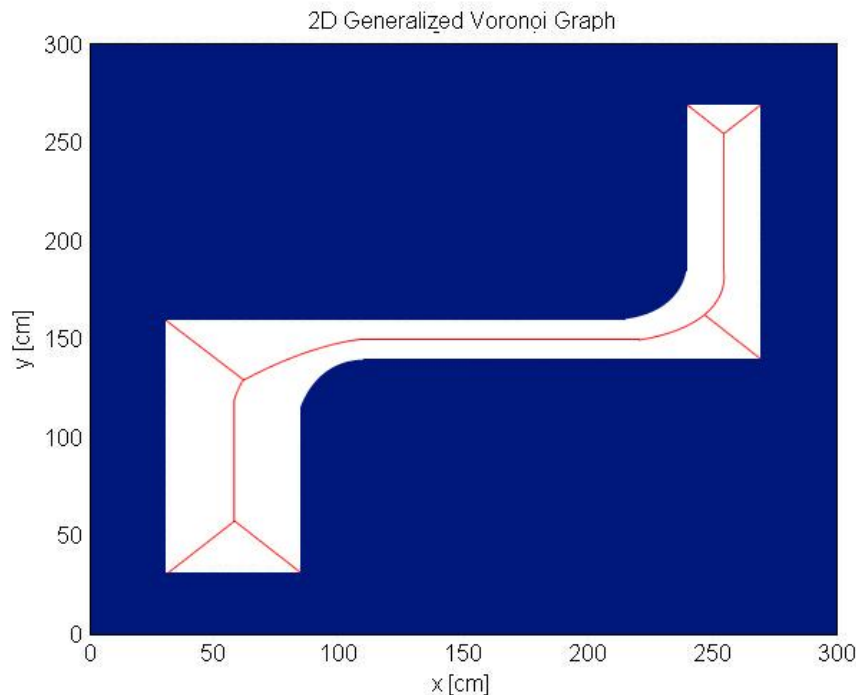
משך הזמן הנדרש לבניית מרחב הקונפיגורציה הדו-מימדי היה $115.39[\text{sec}]$, ומשך הזמן הנדרש לבניית מרחב הקונפיגורציה התלת-מימדי היה כחמישה ימים. ניתן לראות שכבר עבור מרחב קונפיגורציה תלת-מימדי משך הזמן הנדרש לבניית מרחב הקונפיגורציה אינו מעשי, ועבור מרחבים מממדים גבוהים יותר משך הזמן יהיה ארוך פי כמה.

ניתן לראות כי השטחים (הנפחים) החופשיים מצטמצמים בצורה משמעותית. מצד שני ברור כי אי-שימוש במרחב קונפיגורציה מממד מתאים למעשה חוסם קונפיגורציות בהן הכלי כן יכול להימצא. מובן אם כך כי במידת האפשר כדאי להשתמש במרחב קונפיגורציה מממד מלא, אך כאמור, נראה כיצד ניתן לפתור זאת בפרק הבא.

5.5.2 סימולציות לבניית גרף וורנוי הכללי

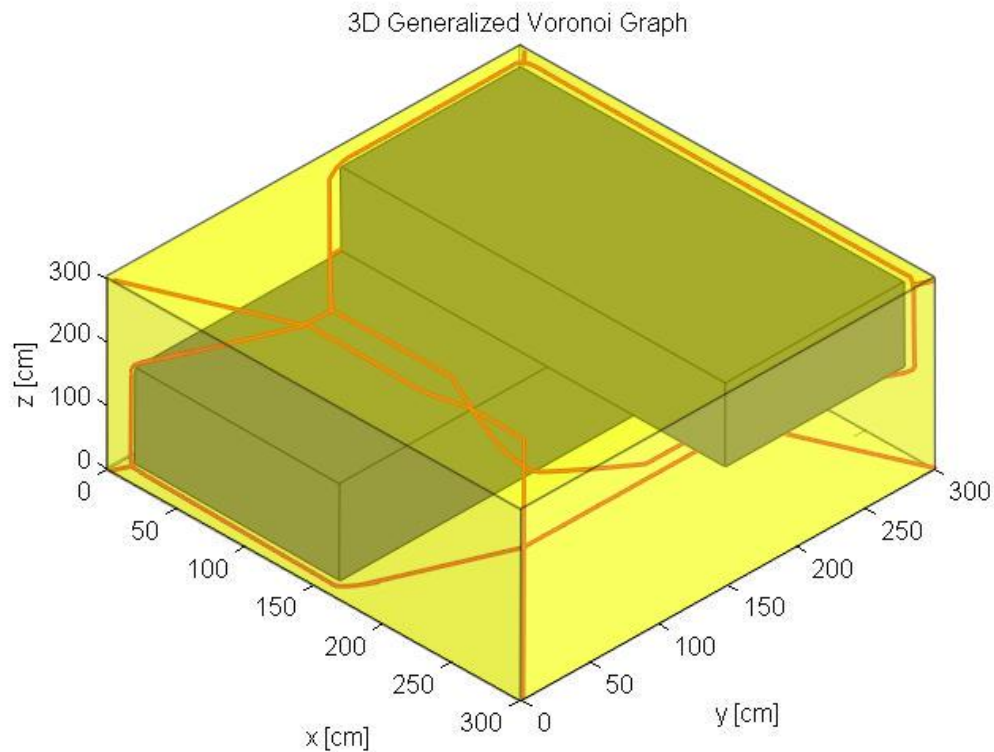
בשלב מאוחר יותר, נרצה לבנות את גרף וורנוי הכללי ישירות על מרחב העבודה ולא על מרחב הקונפיגורציה. לכן, נבצע סימולציות עבור מרחב הקונפיגורציה הדו-מימדי שנבנה לצורך המחשה (איור 5.15), ועל מרחב העבודה התלת-מימדי שהוצג באיור 5.14.

את התוצאה של הפעלת אלגוריתם ה-Brushfire על מרחב הקונפיגורציה הדו-מימדי ניתן לראות באיור 5.17, כאשר בלבן המרחב החופשי, בכחול כהה המכשולים ובאדום גרף וורנוי הכללי.



איור 5.17: גרף וורנוי הכללי שהתקבל באמצעות אלגוריתם ה-Brushfire עבור מרחב הקונפיגורציה הדו-מימדי.

לאחר הפעלת אלגוריתם ה-Brushfire על מרחב העבודה התלת-מימדי, להלן גרף וורנוי הכללי שהתקבל כאשר הנפח הכלוא הוא המרחב החופשי, באפור כהה המכשולים, בצהוב גבולות מרחב העבודה ובאדום גרף וורנוי הכללי:



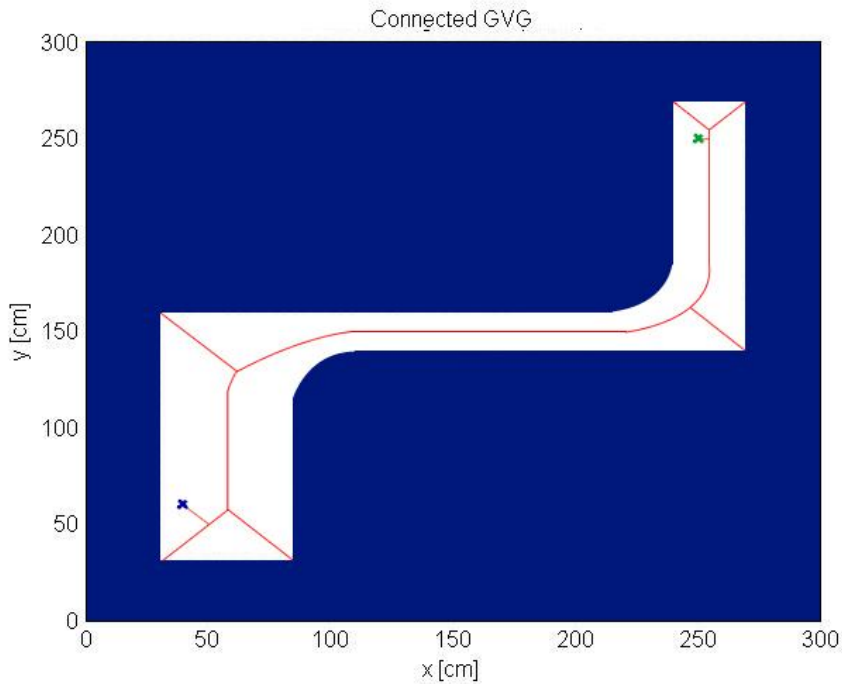
איור 5.18 : גרף וורונוי הכללי שהתקבל באמצעות אלגוריתם ה- Brushfire עבור מרחב העבודה התלת-מימדי.

משך הזמן הנדרש לבניית גרף וורונוי הכללי עבור מרחב הקונפיגורציה הדו-מימדי היה $0.099[\text{sec}]$, ומשך הזמן הנדרש לבניית גרף וורונוי הכללי עבור מרחב העבודה התלת-מימדי היה $193.05[\text{sec}]$.

ניתן לראות כי ע"פ ההגדרה של גרף וורונוי הכללי, הגרף אכן נמצא במרחקים שווים מן המכשולים. כמו כן, מהתבוננות בגרף ברור כי השימוש בגרף וורונוי אכן ייתן לנו מפת דרכים רצויה, שכן השימוש בגרף מבטיח שהכלי יימצא בדיוק בין המכשולים (לפחות תיאורטית) - כלומר הוא רחוק ככל הניתן מהמכשול הקרוב אליו ביותר.

5.5.3 סימולציות למציאת המסלול הקצר ביותר באמצעות האלגוריתם של דייקסטרה

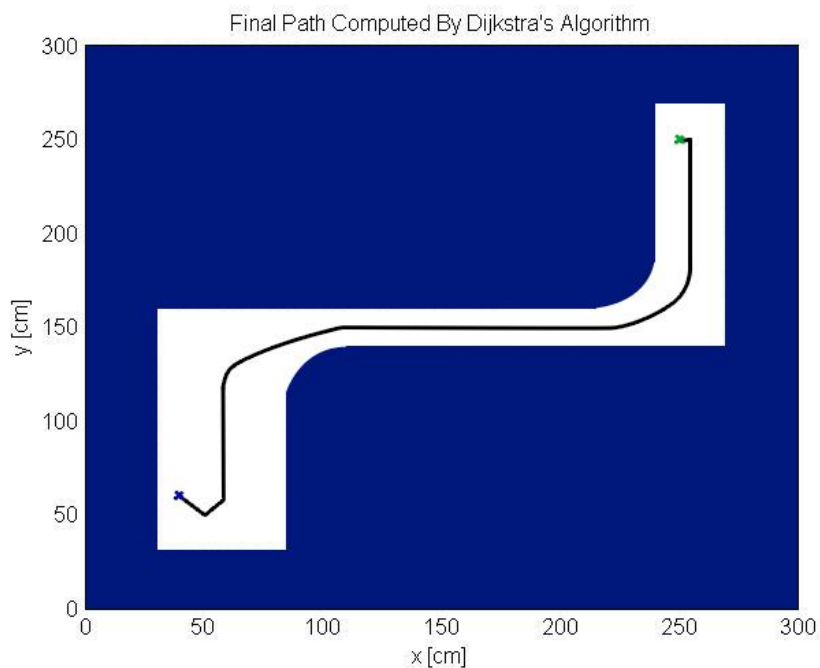
קונפיגורציית המוצא שנבחרה עבוד הסימולציה הדו-מימדית הייתה $x = 60\text{cm}$ ו- $y = 40\text{cm}$, וקונפיגורציית היעד שנבחרה הייתה $x = 250\text{cm}$ ו- $y = 250\text{cm}$. קונפיגורציית המוצא, כמו גם קונפיגורציית היעד, חוברה לגרף וורונוי הכללי באמצעות ישר בין הנקודה המייצגת אותה לבין הנקודה הקרובה ביותר על גרף וורונוי הכללי. להלן תוצאת חיבור זה עבור מצב המוצא והיעד הנתונים עבור מרחב הקונפיגורציה הדו-מימדי, כאשר בכחול מסומן המוצא ובירוק היעד:



איור 5.19: הגרף המתקבל לאחר חיבור המוצא והיעד לגרף וורונוי הכללי עבור מרחב הקונפיגורציה הדו-מימדי.

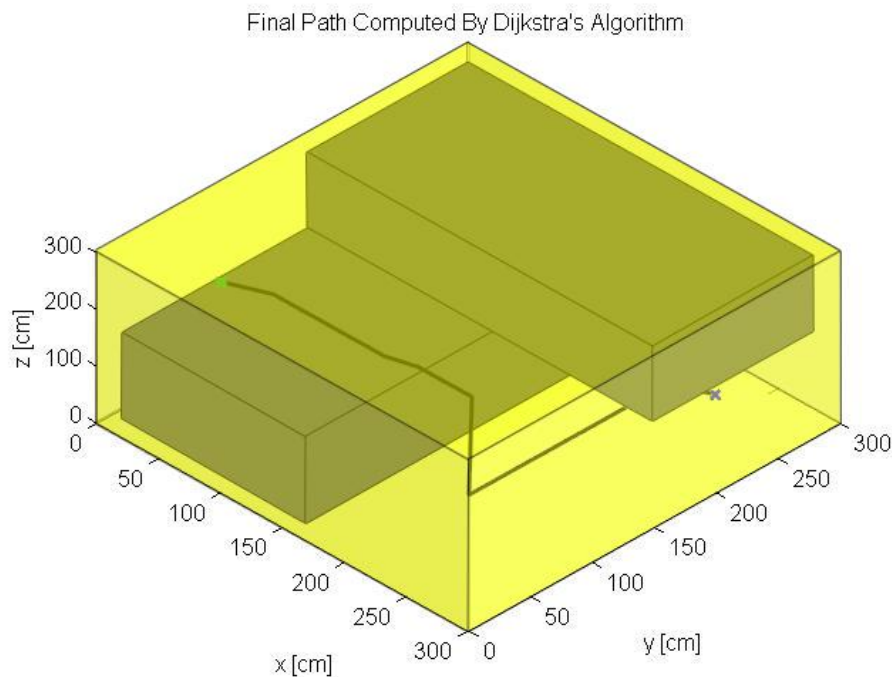
משך הזמן הנדרש לחיבור גרף וורונוי הכללי למוצא וליעד היה $1.06[\text{sec}]$.

את התוצאה של הפעלת האלגוריתם של דייקסטרא למציאת המסלול הקצר על גרף וורונוי הכללי המחובר למוצא וליעד ניתן לראות באיור 5.20, כאשר בכחול מסומן המוצא, בירוק מסומן היעד ובשחור המסלול הקצר שהתקבל. משך הזמן הנדרש למציאת המסלול הקצר ביותר באמצעות האלגוריתם של דייקסטרא היה $10.52[\text{sec}]$.



איור 5.20: המסלול המתקבל לאחר הפעלת האלגוריתם של דייקסטרא עבור מרחב הקונפיגורציה הדו-מימדי.

נקודת המוצא שנבחרה עבור הסימולציה התלת-מימדית הייתה $x = 250\text{cm}$, $y = 250\text{cm}$ ו- $z = 50\text{cm}$. נקודת היעד שנבחרה הייתה $x = 50\text{cm}$, $y = 50\text{cm}$ ו- $z = 250\text{cm}$. נקודות המוצא והיעד חוברו אל גרף וורנוי הכללי בצורה דומה לצורת החיבור עבור המקרה הדו-מימדי. לאחר הפעלת האלגוריתם של דייקסטרא למציאת המסלול הקצר על גרף וורנוי הכללי המחובר למוצא וליעד, להלן המסלול הקצר שהתקבל, כאשר המוצא מסומן בכחול, היעד בירוק והמסלול הקצר בשחור:



איור 5.21: המסלול המתקבל לאחר הפעלת האלגוריתם של דייקסטרא עבור מרחב העבודה התלת-מימדי.

משך הזמן הנדרש לחיבור גרף וורנוי הכללי למוצא וליעד היה $20.84[\text{sec}]$, ומשך הזמן הנדרש למציאת המסלול הקצר ביותר באמצעות האלגוריתם של דייקסטרא היה $200.9[\text{sec}]$.

5.5.4 סיכום תוצאות הסימולציות

מן הסימולציות עבור בניית מרחב הקונפיגורציה ניתן לראות את ההבדל המשמעותי בטופולוגיה בין מרחב העבודה לבין מרחב הקונפיגורציה. אפשר לראות כי המרחב החופשי בו ניתן לתכנן את המסלול במרחב הקונפיגורציה קטן באופן משמעותי ביחס למרחב העבודה החופשי. מכאן ניתן לראות את החשיבות הגדולה שבהתחשבות בקונפיגורציית הכלי בעת תכנון המסלול/תנועה.

כמו כן, ממשך הזמן הנדרש לבניית מרחב הקונפיגורציה התלת-מימדי ברור כי כבר עבור מרחב קונפיגורציה תלת-מימדי בניית מרחב הקונפיגורציה איננה מעשית. כמובן שככל שנגדיל את מימד מרחב הקונפיגורציה יידרש זמן ארוך פי כמה בכדי לבנות אותו. אם כך, ברור כי נאלץ להשתמש בשיטה אחרת לתכנון מסלול/תנועה תוך שימוש במאפיינים שהוצגו עבור מרחב הקונפיגורציה (כגון מכשול קונפיגורציה, מטרה וכו') מבלי לבנות ממש את מרחב הקונפיגורציה.

מהסימולציות לבניית גרף וורונוי הכללי באמצעות אלגוריתם ה - Brushfire ניתן לראות כי הגרפים המתקבלים הן עבור המרחב הדו-מימדי והן עבור המרחב התלת-מימדי מתאימים להגדרה של גרף וורונוי הכללי - כלומר אלגוריתם ה - Brushfire תקין ומניב את התוצאות הדרושות. בנוסף, ניתן לראות באופן מוחשי יותר את היתרון שבשימוש בגרף וורונוי הכללי ביחס למפות דרכים אחרות - המסלול שיתקבל יהיה תמיד במרחק שווה בין המכשולים הקרובים ביותר אליו, כלומר יהיה רחוק ככל הניתן מן המכשול הקרוב ביותר. עובדה זו גורמת לכך שתכנון מסלול באמצעות ה - GVG תניב מסלול בטוח יותר, כלומר לכלי הטיס יהיה יותר מקום לטעות, מאשר שימוש במפות דרכים אחרות. בנוסף, ניתן לראות כי משך הזמן הנדרש לבניית גרף וורונוי הכללי במרחב תלת-מימדי הינו סביר ומתקבל על הדעת - מספר דקות בלבד.

בנוסף, ניתן לראות כי חיבור המוצא והיעד באמצעות קווים ישרים אל הנקודות הקרובות ביותר על גרף הוורונוי הכללי עובד בצורה טובה, כלומר בעת החיבור, קווי החיבור שהוספו תמיד יהיו במרחב החופשי.

מן הסימולציות עבור שימוש באלגוריתם של דייקסטרא למציאת המסלול בקצר ביותר בין המוצא ליעד על גרף וורונוי הכללי ניתן לראות כי פעולת האלגוריתם תקינה, והתוצאה המתקבלת היא אכן המסלול הקצר ביותר. בנוסף, ניתן לראות כי משך הזמן הנדרש לחישוב המסלול הקצר באמצעות האלגוריתם הוא הגיוני ונמשך מספר דקות בודדות.

לסיכום, מפת הדרכים מסוג גרף וורונוי הכללי שנבחרה הינה בחירה מוצלחת. כמו כן, האלגוריתם של דייקסטרא מוצלח גם הוא. לעומת זאת, יש למצוא שיטה אחרת לתכנון מסלול/תנועה מבלי לבנות באופן ממשי את מרחב הקונפיגורציה של כלי הטיס.

6 תכנון תנועה

בפרק זה נציג את שיטת ה- RRT (Rapidly-exploring Random Trees) של Kuffner ו- LaValle [3] לתכנון תנועה עבור סביבה מוכרת. בנוסף, נציג גירסא נוספת שפיתחנו ל- RRT המבוססת על חיפוש סביב ה- GVG שהכרנו בפרק הקודם. בכדי לעשות זאת נסביר תחילה את המושג מרחב מצב ואת הקושי בתכנון במרחב המצב. לאחר מכן נציג את אלגוריתם ה- RRT ואת הגירסא הנוספת שלו, תוך הוכחה כי האלגוריתם שלם הסתברותית (Probabilistically Complete).

החשיבות שבתכנון של תנועה ביחס לתכנון של מסלול, הוא שבעת תכנון מסלול אין כל התחשבות דינמיקה של המערכת ובמגבלות של כניסות הבקרה. בעת תכנון מסלול ההנחה היא כי הרובוט יוכל לעקוב אחר המסלול שתוכנן - ברור כי מצב אידיאלי זה איננו נכון, במיוחד עבור מערכות עם דינמיקה מסובכת וסחיפה (Drift) כתוצאה מגרביטציה לדוגמא. בכדי להמחיש זאת, ניתן לחשוב על קוואדרוטור אשר עוקב אחר המסלול שתוכנן לו, אך עושה זאת במהירות גבוהה, ולפתע ישנה פנייה חדה (במסלול שתוכנן). ברור כי עד שהקוואדרוטור יחזור לעקיבה טובה אחרי המסלול הוא ייסטה ממנו בצורה משמעותית ועלול להתקל במכשול בסבירות גבוהה. לכן, לא מספיק לתכנן את המסלול אותו הרובוט צריך לבצע, אלא ישנו צורך גם לתכנן את המהירויות בהן הוא יעקוב אחר המסלול (תוך התחשבות באילוצים הדינמיים) - זהו תכנון תנועה.

6.1 תכנון במרחב המצב

בכדי לתכנן תנועה במרחב המצב, יש להסביר תחילה מהם מצב ומרחב מצב, כיצד מגדירים מכשול (ואת הבעייתיות שבכך) במרחב המצב ומהי המטרה בזמן תכנון תנועה במושגים של מרחב המצב.

6.1.1 הגדרת מרחב המצב

באלגוריתם ה- RRT תכנון התנועה (או בניית המסלול) מתבצע במרחב המצב (State Space), בניגוד לשיטות הנפוצות והמוכרות יותר, שם בניית המסלול מתבצעת במרחב הקונפיגורציה (בדומה לפרק הקודם). היתרון הגדול בבניית המסלול במרחב המצב הוא שניתן לקחת בחשבון גם את המהירויות, את התאוצות, את הסחיפה ובכלל את הדינמיקה של המערכת (למעשה זהו תכנון תנועה). נגדיר את המצב (State) של רובוט להיות הקונפיגורציה של הרובוט והמהירויות בכיוונים של דרגות החופש שהוגדרו. נסמן כל מצב ב- x ונגדירו (בצורה מתמטית) להיות $x = (q, \dot{q})$. מרחב המצב אם כן, יהיה האוסף של כל המצבים האפשריים של הרובוט והוא מסומן באות X . בדומה למרחב הקונפיגורציה, מצב מסויים יהיה נקודה במרחב המצב. מעצם ההגדרה של מרחב המצב ברור החיסרון העיקרי שבתכנון תנועה במרחב המצב ביחס לתכנון מסלול במרחב הקונפיגורציה - עבור מערכת מסדר שני (עם אילוצים דינמיים), המימד של מרחב המצב יהיה פי שניים מהמימד של מרחב הקונפיגורציה.

מצב המערכת עבור הקוואדרוטור נתון ע"י ווקטור העמודה הבא :

$$\bar{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \\ x_{12} \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ \phi \\ \theta \\ \psi \\ \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (146)$$

6.1.2 אילוצים דיפרנציאליים

כל מערכת מכאנית ניתנת לייצוג באמצעות מערכת של n (או פחות) משוואות דיפרנציאליות מסדר שני, כאשר n הוא המימד של מרחב הקונפיגורציה (לדוגמא משוואה (30)). באמצעות מעבר למרחב המצב (ע"י הורדת סדר של המשוואות הנ"ל) ניתן לקבל מערכת של m ממשוואות מסדר ראשון כאשר $m \leq 2n$. בצורה בלתי מפורשת, מערכת המשוואות תראה כך :

$$\dot{x} = f(x, u) \quad (147)$$

כאשר u הוא ווקטור כניסות הבקרה של המערכת. בצורה מפורשת, מערכת המשוואות עבור הקוואדרוטור תראה כך :

$$\dot{\bar{x}} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \\ \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \\ \dot{x}_{10} \\ \dot{x}_{11} \\ \dot{x}_{12} \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \\ x_{12} \\ (\cos x_4 \sin x_5 \cos x_6 + \sin x_4 \sin x_6) \cdot F_{tot}/m \\ (\cos x_4 \sin x_5 \sin x_6 - \sin x_4 \cos x_6) \cdot F_{tot}/m \\ -g + (\cos x_4 \cos x_5) \cdot F_{tot}/m \\ x_{11}x_{12} \cdot \left(\frac{I_{yy} - I_{zz}}{I_{xx}} \right) + \tau_x / I_{xx} \\ x_{10}x_{12} \cdot \left(\frac{I_{zz} - I_{xx}}{I_{yy}} \right) + \tau_y / I_{yy} \\ x_{10}x_{11} \cdot \left(\frac{I_{xx} - I_{yy}}{I_{zz}} \right) + \tau_z / I_{zz} \end{bmatrix} \quad (148)$$

כאשר את הכניסות $F_{tot}, \tau_x, \tau_y, \tau_z$ נקבל ממערכת הבקרה שהוצגה בפרק 4.

6.1.3 מכשולים במרחב המצב

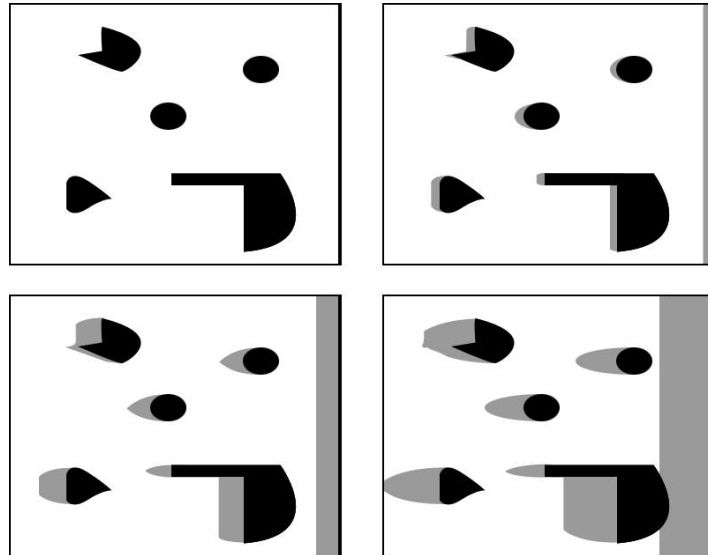
הגדרת מכשולים במרחב הקונפיגורציה היא ברורה, אינטואיטיבית ומתבקשת. הגדרת המכשולים במרחב המצב מסובכת יותר, אך מתבססת על עקרונות דומים. תחילה, נוכל פשוט להשליך את הגדרת המכשולים ממרחב הקונפיגורציה למרחב המצב, כלומר המכשול XO_i במרחב המצב יהיה מורכב ממצבים בהם ישנה התנגשות בין הרובוט לבין המכשול ה- i במרחב העבודה, כלומר:

$$XO_i = \{x \in X \mid R_{occ}(x) \cap WO_i \neq \emptyset\} \quad (149)$$

באופן דומה, נוכל להגדיר את מרחב המצב החופשי כאוסף המצבים בהם אין התנגשות עם מכשול:

$$X_{free} = X \setminus \left(\bigcup_i XO_i \right) \quad (150)$$

ברור כי שימוש בהגדרות אילו לא ייתן כל יתרון לתכנון התנועה במרחב המצב על פני תכנון המסלול במרחב הקונפיגורציה. בכדי לתת יתרון לתכנון במרחב המצב נוכל להגדיר "מצבים מסוכנים". מצב מסוכן יהיה מצב אשר במידה והרובוט נמצא בו, התקלות עם מכשול בפרק זמן קצר תהיה בלתי נמנעת - כלומר ממצב מסוכן, המערכת תגיע בוודאות למצב של התנגשות בעתיד הקרוב. לצורך המחשה נתבונן באיור הבא:



איור 6.1: דוגמא להגדרת מצבים מסוכנים במרחב המצב.

האיור הנ"ל מתאר הערכה למצבים המסוכנים עבור גוף נקודתי הנע על מישור אופקי. נניח כי לגוף יש תאוצה חסומה (כלומר כניסות הבקרה חסומות) ומהירות התחלתית לאורך ציר x בלבד. האיזורים המסומנים בלבן מתארים את המרחב החופשי X_{free} , האיזורים המסומנים בשחור

מתארים את המכשולים $XO_{i=1...k}$ והאיזורים המסומנים באפור הינם הערכה לאיזור המצבים המסוכנים. ברור כי ככל שנגדיל את המהירות ההתחלתית של הגוף, האיזורים האפורים ייגדלו מכיוון שככל שמהירות הגוף לכיוון מכשול תהיה גדולה יותר, כך התנגשות תהיה בלתי נמנעת. כלומר:

$$XR_i = \left\{ x(t) \in X \mid R_{occ} \left(\int_t^{t+\Delta t} f(x(t), u) dt \right) \cap WO_i \neq \emptyset \quad \forall u \right\} \quad (151)$$

ע"י שימוש בהגדרת מצבים מסוכנים נוכל לשפר את הגדרת המרחב החופשי בו ניתן לתכנן את התנועה. נגדיר עתה איזורים של התנגשות בלתי נמנעת (Region of Inevitable Collision - ric). איזור של התנגשות בלתי נמנעת, X_{ric} , יהיה מוגדר להיות האיחוד של האיזור של מכשול המצב ה- i , XO_i , ושל האיזור המסוכן עבור המכשול ה- i :

$$X_{ric} = XO_i \cup XR_i \quad (152)$$

ברור כי כל מצב $x \in X_{ric_{i=1...k}}$ אינו רצוי שכן הוא מצביע על כך שישנה התנגשות או שתהיה התנגשות. לכן, נגדיר את מרחב המצב החופשי להיות אוסף המצבים בהם אין התנגשות ואין התנגשות בלתי נמנעת:

$$X_{free} = X \setminus \left(\bigcup_i X_{ric_i} \right) \quad (153)$$

מעצם ההגדרה (ומהתבוננות באיור 6.1) ברור כי שימוש בהגדרה זו תשנה באופן מהותי את הטופולוגיה של מרחב המצב. ההבדל עשוי להיות כ"כ קיצוני כך שעבור שימוש בהגדרה הקלאסית (כלומר תכנון מסלול) נוכל למצוא פתרון, בעוד שעבור שימוש בהגדרה (153) לא תהיה תנועה כזו שתוכל להביא לפתרון הבעיה. עובדה זו מראה בבירור כי תכנון בגישה הקלאסית, דהיינו תכנון מסלול בנפרד ותכנון בקר לעקיבה בנפרד, לא תעבוד במקרים רבים, במיוחד עבור מצבים בהם ישנה סחיפה (כמו גרביטציה במקרה שלנו) והגבלה על כיוון וגודל הכניסות (במקרה שלנו הכוח שיכול להפעיל כל מנוע חסום וכיוונו כלפי מעלה בלבד). בנוסף, ברור כי עבור רובן המכריע של המערכות לא ניתן לבצע אבחנה בין מצבים מסוכנים לבין מצבים חופשיים בצורה מדויקת. מכאן ניתן לראות את הקושי הגדול שבתכנון תנועה עבור מערכות עם אילוצים דינמיים כדוגמת הקוואדרוטור. שימוש באלגוריתם ה- RRT פותר בעיות אילו מפני שאין צורך לבנות בצורה מפורשת את כל מרחב המצב - נרחיב על כך בהמשך.

6.1.4 מטרות תכנון התנועה במרחב המצב

בצורה דומה לזו שהגדרנו עבור מרחב הקונפיגורציה, נוכל להגדיר את המטרה עבור תכנון תנועה במרחב המצב. בעת תכנון תנועה במרחב המצב, המטרה היא למצוא העתקה שהיא פונקציה של הזמן $\tau : [0, T] \rightarrow X_{free}$ כך ש- $\tau(0) = x_{start}$ ו- $\tau(T) = x_{goal}$, כאשר x_{start} הינו מצב המוצא ו- x_{goal} הינו מצב המוצא ו- x_{goal} .

x_{goal} הינו מצב היעד. פתרון נוסף יכול להיות העתקה הנותנת רצף של כניסות $u: [0, T] \rightarrow U$ במקום רצף של מצבים, כך ששימוש ברצף הכניסות הנ"ל יביא את המערכת מהמצב ההתחלתי למצב הרצוי תוך המנעות ממכשולים - כלומר בקרה בחוג פתוח.

ברור כי עשויים להיות אינספור פתרונות אפשריים לתכנון התנועה. את הפתרון הסופי ניתן לברור מתוך הפתרונות הנתונים על-סמך קריטריונים שונים כגון מהירות הגעה, מהירות מקסימלית, אנרגיה מושקעת, משך ריצת אלגוריתם הפתרון וכו'.

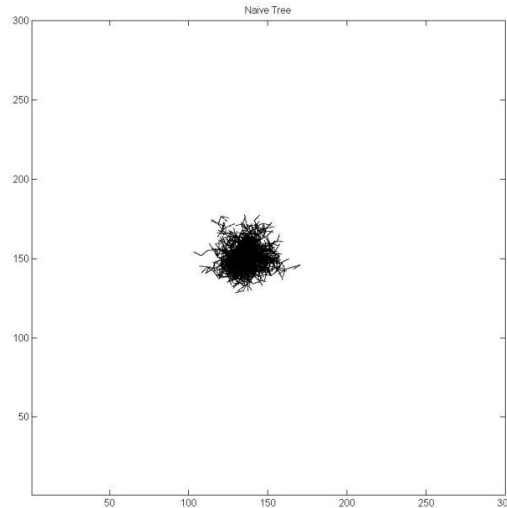
6.2 אלגוריתם ה - RRT

אלגוריתם ה - RRT (Rapidly-exploring Random Trees) הוא אלגוריתם תכנון תנועה (או מסלול) רנדומלי אשר פותח ע"י Kuffner ו- LaValle [3] עבור מערכות עם אילוצים קינו-דינמיים, ומתאים במיוחד למערכות עם אילוצים דפירנציאליים מסדר שני. הדרישה לשימוש באלגוריתם היא כי עבור כל מצב של המערכת ועבור כל כניסה נתונה, ניתן לבצע סימולציה לקבלת מצב המערכת לאחר פרק זמן נתון. במקרה שלנו נוכל להשתמש ב- (148) ולבצע התקדמות בזמן באמצעות פתרון נומרי כלשהו (לדוגמא Runge-Kutta) מהמצב הנתון עבור פרק הזמן הרצוי. נסביר עתה את דרך פעולת האלגוריתם, נוכיח כי האלגוריתם שלם הסתברותית (Probabilistically Complete) וניתן הערכה לכמות החישובים אשר מבצע האלגוריתם.

6.2.1 פעולת האלגוריתם

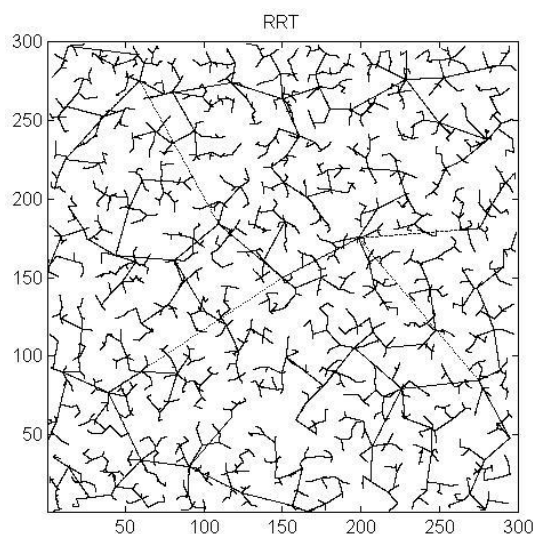
מטעמי נוחות, נניח כי ההתקדמות הנומרית בזמן של (148) עבור מצב נתון x_k ועבור כניסה נתונה u_k נתונה ע"י $x_{k+1} = f(x_k, u_k)$. נסביר תחילה מה עשויה להיות דרך הפעולה באלגוריתמים רנדומליים אינטואיטיביים אחרים, לאחר מכן נסביר את פעולת ה - RRT ונראה את יתרונו הגדול על פני שימוש באלגוריתמים שונים.

בכדי להבהיר את היתרון המובהק של אלגוריתם ה - RRT נתבונן בדוגמא פשוטה של גוף נקודתי הנע במרחב דו-מימדי. גישה רנדומלית אחת עשויה להיות כדלהלן: עבור מצב התחלתי נתון x_{start} בחר כניסה u_k רנדומלית כלשהי והפעלת אותה מהמצב הנתון עבור פרק זמן קצר מסויים Δt . לאחר הפעלת הכניסה שנבחרה וההתקדמות הנומרית בזמן, הוסף את הצומת החדש x_k אליו הגיע הגוף לגרף בתנאי שאין התנגשות עם מכשול בדרך. עתה בחר צומת רנדומלי (מצב רנדומלי) מהגרף, בחר כניסה רנדומלית u_k , התקדם נומרית בזמן והוסף את הצומת החדש וחוזר חלילה עד להגעה למצב הרצוי x_{goal} . ניתן לדמות זאת לעץ הנשתל במצב x_{start} וצומח במרחב המצב. מעתה נתייחס לגישה זו בשם "עץ רנדומלי נאיבי". הגישה הנ"ל היא אמנם רנדומלית, אך היא בעלת משיכה חזקה מאוד למצבים בהם המערכת כבר הייתה. ניתן לראות זאת באיור הבא:



איור 6.2 : דוגמא לעץ רנדומלי נאיבי עם 2000 צמתים.

נסביר עתה את אלגוריתם ה- RRT ונראה את יתרונו הגדול ביחס לשיטות רנדומליות אחרות. בשלב הראשון "נשתול" את העץ הרנדומלי שלנו במצב המוצא x_{start} . כעת, נבחר מצב רנדומלי x_{random} כלשהו ממרחב המצב החופשי. מתוך העץ הקיים, נחפש את הצומת x_k הקרוב ביותר ל- x_{random} . לאחר מציאת הצומת הקרוב ביותר, נפעיל את הכניסה u_k , מתוך תחום הכניסות המותרות, שתביא למזעור (במצב אידיאלי - מזעור מקסימלי) של המרחק שבין x_k ובין x_{random} . נתקדם בצורה נומרית בזמן עבור המצב הנתון ועבור הכניסה הנתונה, עבור פרק זמן נתון Δt , כך שנקבל את הצומת החדש $x_{k+1} = f(x_k, u_k)$. במידה ואין התנגשות עם מכשול בדרך לצומת החדש x_{k+1} נוסיף את המצב הני"ל כצומת לעץ שלנו ונחזור על התהליך הני"ל עד להגעה ליעד הרצוי x_{goal} (או למצב קרוב לו), או עד שבוצעו מספר מוגדר של חזרות. לפני שנמשיך, נתבונן בתוצאה של שימוש באלגוריתם ה- RRT עבור הבעיה שנפתרה ע"י שימוש בעץ הנאיבי (איור 6.2):



איור 6.3 : דוגמא ל- RRT עם 2000 צמתים.

השוואה בין התוצאות של שני האלגוריתמים השונים מראה את היתרון הברור שיש לאלגוריתם ה-RRT. שימוש באלגוריתם זה מוביל לכיסוי מהיר ואחיד יחסית של המרחב, בעוד האלגוריתם הנאיבי מתפשט במרחב באיטיות גדולה מאוד. הסיבה ליתרון של אלגוריתם ה-RRT היא העובדה שלאלגוריתם הנ"ל יש משיכה חזקה לעבור האיזורים ריקים שטרם נבחנו, בעוד שבאלגוריתם הנאיבי ישנה משיכה חזקה לעבר מצבים שנבחנו בעבר. כלומר, העץ באלגוריתם ה-RRT "נמשך" כך שיכסה במהירות (Rapidly) את המרחב - ומכאן נגזר שמו.

נציג עתה את הפסאודו-קוד המתאר את פעולת אלגוריתם ה-RRT ולאחר מכן נסבירו בצורה מפורטת. להלן הפסאודו-קוד עבור אלגוריתם ה-RRT:

```

1  function RRT( $x_{start}, x_{goal}, Workspace$ ):
2       $T_{initial} = x_{start}$  ;
3      for  $k = 1$  to  $K$  :
4           $x_{random} = \text{Random\_State}(Workspace)$  ;
5           $\text{Extend}(T, x_{random})$  ;
6          if  $T$  reached  $x_{goal}$ :
7               $Path = \text{Find\_Path}(T)$  ;
8              return  $Path$  ;
9      return  $T$  ;

```

איור 6.4 : הפסאודו-קוד לתכנון תנועה באמצעות אלגוריתם ה-RRT.

כאשר הפונקציה $\text{Extend}(T, x)$ נתונה ע"י הפסאודו-קוד הבא :

```

1  function  $\text{Extend}(T, x)$  :
2       $x_{near} = \text{Nearest\_Neighbor}(x, T)$  ;
3      if  $\text{New\_State}(x, x_{near}, x_{new}, u_{new})$ :
4           $\text{Add\_Vertex}(T, x_{new})$  ;
5           $\text{Add\_Edge}(x_{near}, x_{new}, u_{new})$  ;
6          if  $x_{new} = x$  :
7              return Reached ;
8          else
9              return Advanced ;
10     return Trapped ;

```

איור 6.5 : הפסאודו-קוד עבור הפונקציה $\text{Extend}(T, x)$.

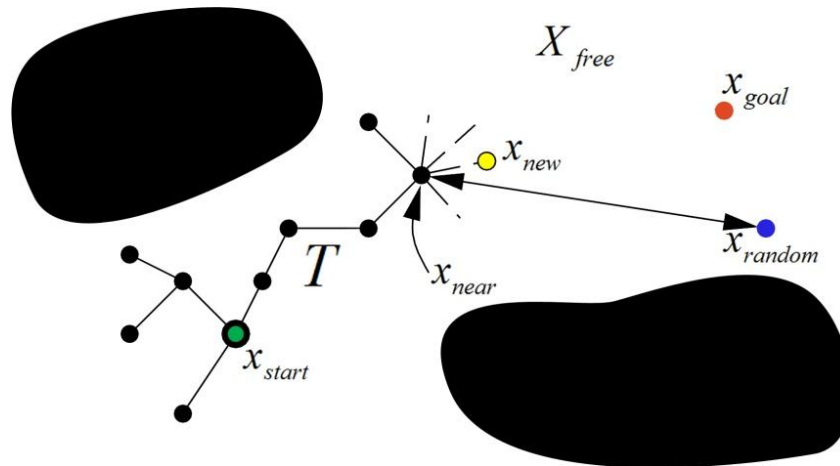
תחילה, נשתול (או נאתחל) את העץ שלנו להיות מצב המוצא x_{start} . עתה, עבור מספר מוגדר של איטרציות ננסה להתקדם בכל איטרציה בצורה הבאה: באמצעות הפונקציה

$Random_State(Workspace)$ נבחר את x_{random} באמצעות מנוע המייצר מצבים רנדומליים במרחב המצב הנתון עם התפלגות כלשהי, ולאחר מכן ננסה להתקדם בכיוון המצב החדש לפי עקרונות האלגוריתם. ניסיון ההתקדמות (או ההתקדמות עצמה במצב הצלחה) יעשה באמצעות הפונקציה $Extend(T, x)$.

לפי פעולת האלגוריתם, ההתקדמות בכיוון המצב הרנדומלי x_{random} תעשה מהצומת הקרוב ביותר אליו אשר כבר נמצא בעץ. לכן, באמצעות הפונקציה $Nearest_Neighbor(x, T)$ נבחר את הצומת הקרוב ביותר מהעץ למצב הרנדומלי. חשוב לציין כי מידת הקירבה בין צומת אחד לצומת אחר תיקבע ע"פ פונקציית מרחק כלשהי - בחירת פונקציית המרחק עשויה להיות סוגיה בפני עצמה, ובחירה בפונקצייה מסויימת עשויה לזרז משמעותית את פעולת האלגוריתם ביחס לפונקציית מרחק שונה. כעת, לאחר שנמצא הצומת הקרוב ביותר x_{near} מהעץ ננסה להתקדם כלפי הצומת הרנדומלי בצורה שתמזער את המרחק ביניהם (מיזעור מקסימלי יהיה אופטימלי). התקדמות זו, מהצומת x_{near} לכיוון הצומת x_{random} , תעשה באמצעות הפונקציה $New_State(x, x_{new}, x_{random}, u_{new})$.

לפונקציה $New_State(x, x_{new}, x_{random}, u_{new})$ מספר תפקידים חשובים. ראשית, על הפונקציה למצוא את הכניסה שתמזער את המרחק בצורה הטובה ביותר. במידה ומדובר בכניסות בדידות ניתן לנסות את כלל הכניסות ולבסוף לבחור את הכניסה הטובה ביותר, ובמידה ולא מדובר בכניסות בדידות ניתן לבחור את הכניסות על סמך אנליזה מוקדמת. במקרה שלנו השתמשנו במערכת הבקרה אשר הוצגה בפרק 4. לאחר בחירת הכניסה (או הכניסות) הפונקציה תתקדם נומרית ב- Δt (פרק זמן קצר נתון) ע"פ משוואת הדינמיקה של המערכת, באמצעות הסכימה הנומרית שנבחרה לפתרון. את המצב אליו הגיעה המערכת לאחר ההתקדמות בזמן נסמן ב- x_{new} . כעת, הפונקציה תבדוק אם אירעה התנגשות עם מכשול במהלך הדרך שעברה המערכת בין המצב x_{near} למצב x_{new} . במידה וישנה התנגשות, הפונקציה תודיע על כך ותעבור לאיטרציה הבאה. במידה ואין התנגשות ישנן שתי אפשרויות. האפשרות הראשונה היא שהמערכת הגיעה למצב x_{random} (או קרוב אליו) - הפונקציה תודיע על הגעה, והאפשרות השנייה היא שהמערכת לא הגיעה למצב x_{random} אלא רק התקדמה בכיוונו - הפונקציה תודיע על התקדמות. בכל מקרה, במידה ולא אירעה התנגשות, נוסיף את המצב x_{new} כצומת נוספת לעץ הקיים. בנוסף, ניתן לשמור את רצף הכניסות בהן השתמשנו במהלך האלגוריתם בכדי להשתמש בהן עבור בקרה בחוג פתוח. דוגמא להתקדמות באמצעות אלגוריתם ה- RRT ניתן לראות באיור 6.6.

חשוב לציין כי עבור מערכת עם אילוצים דיפרנציאליים, בדומה לקוואדרוטור, יש לייצר גרף מכון ע"י שמירה של אינדקס הצומת ממנה יצאנו והאינדס אליו הגענו, וזאת משום שעבור מערכת עם אילוצים דינמיים ישנה חשיבות מכרעת לכיוון ההתקדמות, ולרוב לא ניתן להחזיר את המערכת חזרה בדיוק באותה הדרך דרכה הגענו.



איור 6.6 : התקדמות המערכת בכיוון המצב x_{random} .

תנאי הסיום של האלגוריתם הוא אחד משני מקרים. תנאי הסיום הראשון הוא שהאלגוריתם הצליח לחשב תנועה שתביא למצב היעד הרצוי, כלומר מצב היעד (או מצב הנמצא בסביבה הקרובה לו) הוסף כצומת אל העץ וניתן לקבל את התנועה (המסלול) הרצוי ע"י חזרה מן הצומת הסופי אל צומת המוצא. תנאי הסיום הוא הגעה למספר האיטרציות המוגדר ללא מציאת תנועה שתפתור את הבעיה. במקרה כזה ניתן להחליט על ביצוע של מספר מוגדר של איטרציות נוספות ולהמשיך לחפש פתרון, או שניתן להגיע למסקנה כי לא קיים פתרון לבעיה. תנאי העצירה השני הוא התנאי הבעייתי עבור כל השיטות הרנדומליות לתכנון תנועה - במידה וקיים פתרון נקבל אותו, אך במידה והפתרון אינו קיים, לעולם לא נדע זאת בוודאות.

6.2.2 הוכחת שלמות הסתברותית

עבור כל שיטה נתונה למציאת מסלול שאיננה רנדומלית או סטטיסטית, ניתן להוכיח כי אם קיים פתרון האלגוריתם ייתן את הפתרון, וכי אם הפתרון לא קיים האלגוריתם יודיע חד-משמעית כי אין פתרון (בדומה לאלגוריתם של דייקסטרא מתת-הפרק 5.4). לעומת זאת, בשיטות רנדומליות כדוגמת אלגוריתם ה-RRT, לעולם לא נוכל לקבל תשובה חד-משמעית אם הפתרון לא קיים במידה וזהו המצב, אך נוכל להראות כי כשמספר האיטרציות של האלגוריתם שואף לאינסוף, ההסתברות כי נגיע לפתרון (במידה והוא קיים) שואפת ל-1. התכנסות מעין זו של אלגוריתם נתון לפתרון, כאשר מספר האיטרציות שואף לאינסוף נקראת "שלמות הסתברותית" (Probabilistically Complete). נרצה להראות כי האלגוריתם שלם הסתברותית. בכדי לעשות זאת נרצה להוכיח את הטענה הבאה:

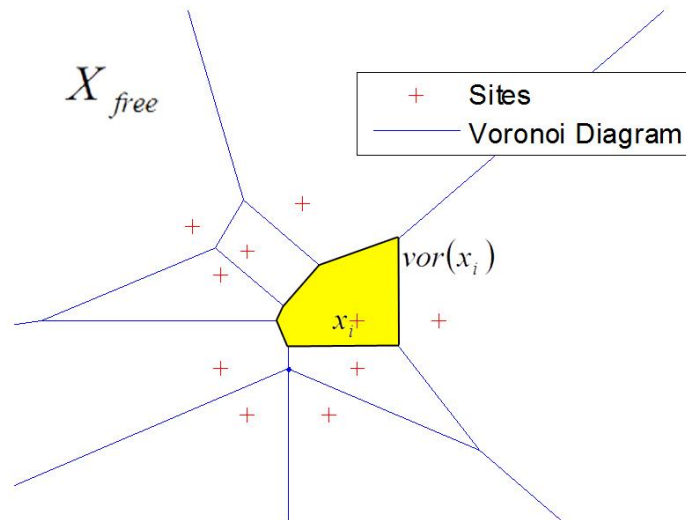
טענה:

ההסתברות שאלגוריתם ה-RRT אשר מאותחל ב- x_{start} יכיל את x_{goal} כצומת שואפת ל-1 כאשר מספר הצמתים שואף לאינסוף, במידה והפתרון קיים.

הוכחה:

נוכיח טענה זו באמצעות אינדוקציה על i . נניח כי העץ מכיל את המצב x_i כצומת לאחר מספר סופי של איטרציות. נתבונן בדיאגרמת הוורוני של הצמתים אשר שייכים לעץ (לצורך המחשה

ניתן להתבונן באיור 6.7). עבור הצומת x_i ועבור קבוע חיובי כלשהו c_1 נוכל לרשום כי $\mu(vor(x_i)) > c_1$, כאשר $\mu(vor(x_i))$ מסמל את השטח של איזור הוורוני השייך לצומת x_i . עתה, אם המצב הרנדומלי הבא ייפול בתוך $vor(x_i)$, הצומת x_i יהיה המצב הקרוב ביותר למצב הרנדומלי, ועליו תופעל הכניסה. כלומר, לצומת x_i יש את ההסתברות $\mu(vor(x_i))/\mu(X_{free})$ להיבחר עבור כל איטרציה. לשם נוחות, נניח עתה כי הכניסות רנדומליות. את ההסתברות שהכניסה הנכונה תופעל נסמן ב- c_2 (כמובן ש- c_2 תלוי ב- c_1), כאשר הכניסה הנכונה היא הכניסה שתביא את המערכת לצומת הרנדומלי מהצומת x_i . מאחר וההסתברות לבחירת מצב באיזור הוורוני של x_i היא קבועה עד לבחירת מצב באיזור וורוני זה, ההסתברות כי איזור זה ייבחר תהיה 1 עבור אינסוף איטרציות. לאחר מכן, ההסתברות לבחירת הכניסה הנכונה תהיה c_2 בכל איטרציה בה נבחר איזור הוורוני של x_i , ולכן ההסתברות להוספת המצב הבא ברצף הפתרון כצומת לעץ תהיה 1 עבור אינסוף איטרציות. בצורה זו נתקדם לאורך מסלול הפתרון (במידה וקיים) מן הצומת הראשון x_{start} ועד לצומת x_k , עד לשלב הסופי בו $x_{goal} \approx x_{k+1}$.



איור 6.7: דיאגרמת וורוני להמחשת הוכחת השלמות ההסתברותית.

6.2.3 הערכה לכמות האיטרציות הדרושה

ננסה עתה לתת הערכה לכמות האיטרציות הדרושה בכדי להגיע מצומת המוצא אל מצב היעד הרצוי. נציין כי חישוב בפועל של הקבועים והמספרים שנגדיר הוא כמעט בלתי-אפשרי ועשוי להיות מסובך יותר מפתרון אנליטי לבעיית תכנון התנועה, אך בכל מקרה ייתן לנו אומדן כלשהו. נגדיר תחילה את המושג "אגן" (Basin).

אגן (Basin/Attractor) יהיה תת-המרחב $B \subseteq X$ של מרחב המצב של מערכת אשר הדינמיקה שלה נתונה ע"י (147), אשר עונה על התנאים הבאים:

- אם המצב $x(t) \in B$ שייך לאגן B , אזי גם לאחר התקדמות בזמן ב- $\Delta t > 0$, המצב

$$x(t + \Delta t) = \int_t^{t+\Delta t} f(x(t), u) dt$$

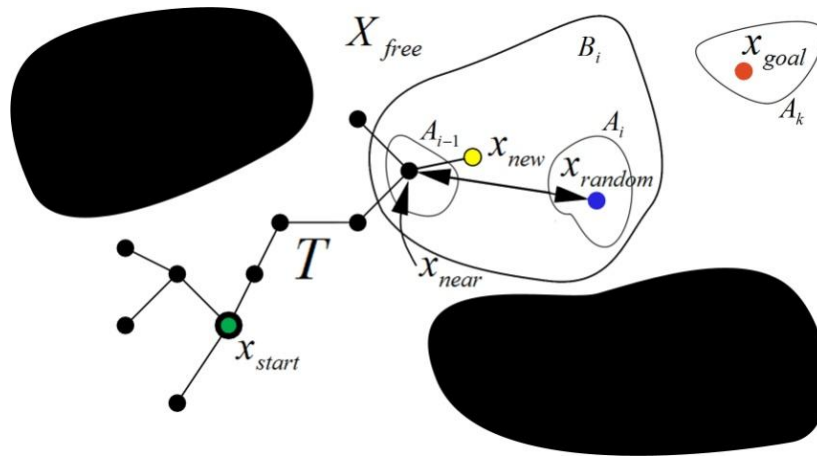
יהיה שייך לאגן B .

- קיימת סביבה A בתוך B ($A \subseteq B$) כך שעבור מצב כלשהו $x(t) \in B$ המערכת תימשך לסביבה A , כך ש- $x(t + \Delta t) \in A$ עבור $\Delta t > T$ כאשר T קבוע חיובי כלשהו. סביבה זו נקראת "אגן משיכה" (Basin of Attraction).

נגדיר את $A = \{A_0, A_1, \dots, A_k\}$ להיות רצף של תתי-מרחב של X , אליו נתייחס כאל "רצף המשיכה". נגדיר את A_0 להיות $A_0 = x_{start}$. את שאר תת-המרחבים נבחר כך שלכל A_i קיים אגן $B_i \subseteq X$ אשר מקיים את התנאים הבאים:

- עבור שלושה מצבים כלשהם $x \in A_{i-1}$, $y \in A_i$, $z \in X \setminus B_i$, המטריקה (פונקציית המרחק) ρ שנבחרה תיתן $\rho(x, y) < \rho(x, z)$.
 - עבור כל מצב $x \in B$ קיים רצף סופי של כניסות (המתקבלים מהפונקציה Extend)
- $$\{u_1, u_2, \dots, u_m\}, \text{ כך שהאלגוריתם יגיע למצב בתוך } A_i \subseteq B_i.$$

האיבר האחרון ברצף המשיכה יהיה $A_k \approx x_{goal}$. בצורה אינטואיטיבית, המשמעות של התנאי הראשון הוא כי בעת יצירת המצב הרנדומלי $y \in A_i$, המצב $x \in A_{i-1}$ יימשך לעבר תת-המרחב A_i משום שייבחר כשכן הקרוב ביותר ע"י המטריקה. אמנם ע"פ ההגדרה, נראה כי המערכת לא תוכל לצאת מ- A_{i-1} , אך ע"י שינוי המצב אליו תרצה להגיע המערכת אשר נמצא מחוץ ל- A_{i-1} אנו גורמים לשינוי במערכת הבקרה ועקב כך לשינוי בתגובת המערכת - תגובה שתוביל ליציאה מ- A_{i-1} ומשיכה ל- A_i . משמעות התנאי השני היא כי לאחר כניסה לאגן B_i האלגוריתם יימשך כלפי תת-המרחב A_i - למעשה תת-המרחב A_i הוא אגן משיכה בתוך האגן B_i . כלומר האגן B_i הוא תת-מרחב בטוח בו המערכת תישאף בסופו של דבר להגיע אל A_i . לצורך המחשה ניתן להתבונן באיור 6.8. בצורה זו המערכת תימשך מתת-המרחב $A_0 = x_{start}$ אל האגן B_1 , משם המערכת תימשך לתת-המרחב A_1 , לאחר מכן תימשך המערכת אל האגן B_2 וחוזר חלילה עד שהמערכת תגיע אל תת-המרחב $A_k \approx x_{goal}$. למעשה המצבים בתוך רצף המשיכה ייתנו לנו בסופו של דבר את תנועת הפתרון.



איור 6.8 : המחשת אגן ואגני משיכה.

את רצף המשיכה יש לבחור כך שכל A_i יהיה גדול ככל הניתן ושאוורכו של רצף המשיכה יהיה קצר ככל האפשר. מכאן נובע כי אם מרחב העבודה כולל מעברים צרים, אזי רצף המשיכה יהיה ארוך יותר ותתי-המרחבים שבו יהיו קטנים יותר. מכאן ברור כי עבור מרחבי עבודה מסוג זה משך הפתרון יהיה ארוך בהרבה. חסם תחתון להסתברות p כי מצב רנדומלי ייפול בתוך תת-מרחב A_i כלשהו נתון ע"י:

$$p = \min_i \{ \mu(A_i) / \mu(X_{free}) \} \quad (154)$$

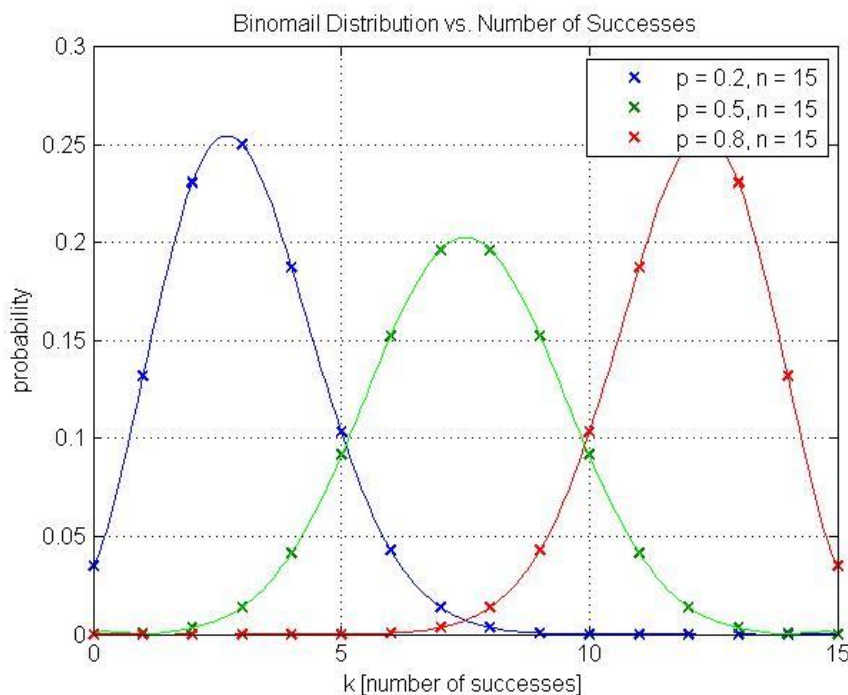
נרצה עתה להראות כי במידה וקיים רצף משיכה באורך k מספר האיטרציות הצפוי (בעל ההסתברות הגבוהה ביותר להתרחשות) בכדי לחבר את x_{start} ל- x_{goal} הוא לכל היותר k/p .

במידה וצומת מתוך העץ T שוכן בתוך אגן המשיכה A_{i-1} ומצב רנדומלי כלשהו x שוכן בתוך אגן המשיכה A_i , אזי העץ יתחבר בסופו של דבר למצב בתוך A_i (ראה הסבר על דרך בחירת רצף ההתכנסות). בכל איטרציה, ההסתברות שהמצב הרנדומלי ייפול בתוך A_i תהיה p . כלומר, אם A_{i-1} מכיל מצב אשר שייך כצומת לעץ T , אזי ההסתברות כי העץ יכיל מצב השייך ל- A_i תהיה p . במקרה הגרוע ביותר ניתן להסתכל על האיטרציות כעל ניסיונות ברנולי בהן ההסתברות להצלחה תהיה p . תנועת הפתרון תתקבל לאחר k ניסיונות מוצלחים משום שכל איטרציה מוצלחת מקדמת את העץ מ- A_{i-1} אל A_i , כאשר $|A| = k$.

נסמן ב- C_1, C_2, \dots, C_n סדרה של n ניסיונות זהים ובלתי תלויים סטטיסטית (ניסיונות ברנולי) עם הסתברות p להצלחה. המשתנה המקרי $C = C_1 + C_2 + \dots + C_n$ מציין את מספר ההצלחות לאחר n ניסיונות. משום שכל C_i הוא בעלת התפלגות ברנולי C יהיה בעל התפלגות של משתנה מקרי בינומיאלי. ע"פ ההתפלגות של המשתנה מקרי בינומיאלי, ההסתברות להצלחה ב- k ניסיונות תהיה נתונה ע"י:

$$\frac{n!}{(n-k)! \cdot k!} p^k \cdot (1-p)^{n-k} \quad (155)$$

נתבונן עתה בגרף 6.1 המציג את פילוג ההסתברות לקבלת מספר שונה של הצלחות עבור הסתברויות p שונות. כפי שניתן לראות, עבור כל אחד מן המקרים המוצגים בגרף, ההסתברות הגבוהה ביותר למספר הצלחות מתקבלת עבור מספר הצלחות השווה ל- $n \cdot p$. כזכור, נרצה לקבל k נסיונות מוצלחים ולכן ההסתברות הגבוהה ביותר לקבלת מספר כזה של הצלחות עבור p נתון תתקבל כאשר נבצע איטרציות כמספר ההצלחות לחלק בהסתברות, כלומר הצפי להצלחה בהסתברות בהתפלגות של משתנה מקרי בינומיאלי (וגם חסם עליון על ההסתברות למציאה מוצלחת של הפתרון) היא k/p .



גרף 6.1: התפלגות בינומיאלית כתלות במספר ההצלחות עבור הסתברויות שונות.

6.3 אלגוריתם ה- RRT

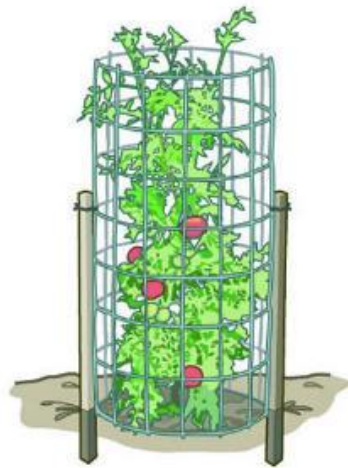
לאחר הבנת היתרונות שבשימוש באלגוריתם ה- RRT, יתרונות כדוגמת אי-הצורך בבניית מרחב קונפיגורציה מלא (או מרחב מצב מלא) ועם זאת התחשבות באילוצים כתוצאה מהדינמיקה של המערכת ובחסמים עבור גודל הכניסות, ולאחר הבנת החסרונות העיקריים של האלגוריתם, חסרונות כדוגמת אי-שלמות האלגוריתם (במידה ולא קיים פתרון לא נדע זאת בוודאות), מצבים נוצרים באופן רנדומלי לגמרי ללא התחשבות כלל במרחב העבודה ועבור מרחבי עבודה עם מעברים צרים משך זמן הריצה של האלגוריתם יהיה ארוך ביותר, ננסה לשפר את ביצועי האלגוריתם. לאלגוריתם המשופר הוחלט לקרוא RRT - ראשי תיבות של Rapidly-exploring Random Tomato Tree, אותם נסביר מיד.

6.3.1 פעולת האלגוריתם

המחשבה העומדת מאחורי השינויים אשר נעשו באלגוריתם הני"ל ביחס לאלגוריתם המקורי היא כי המנוע (הפונקציה) המייצר מצבים רנדומליים עושה זאת ללא כל התחשבות במאפיינים של מרחב העבודה. לכן, ננסה לייצר מצבים בצורה מעט חכמה ביותר.

ישנן גישות בהן הפונקציה המייצרת מצבים רנדומליים מייצרת אחת למספר מוגדר של נסיונות את המצב x_{goal} מתוך המחשבה כי פעולה זו תגרום ל"צמיחה" של העץ בכיוון הפתרון. עם הרעיון של ייצור מצבים בצורה חכמה יותר בכדי לגרום לצמיחת העץ בכיוון הרצוי מולנו נתבונן בשתי צורות עיקריות לגידול עגבניות.

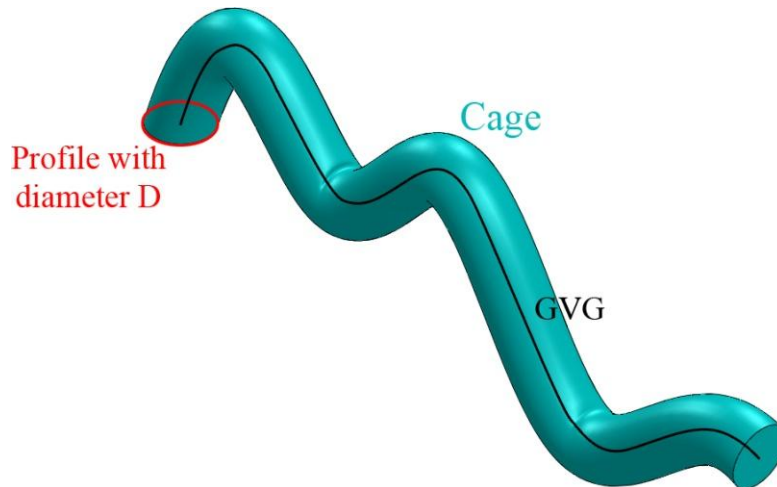
הגישה הפשוטה לגידול עגבניות היא גידול בצורת שיח עגבניות פשוט ללא כל פעולה מיוחדת מלבד השקיה. בגידול כזה השיח מניב את כל הפירות שלו בערך באותה תקופת זמן, ולאחר שהפירות (עגבניות) מתפתחים השיח נובל ולמעשה מת. עובדה זו גורמת לכך שבעת גידול עגבניות נרצה כי השיח ייגדל כלפי מעלה בכדי שיהיה גבוה, ובצורה זו הוא יניב את פירותיו בצורה ממושכת ולאורך זמן, כך שנקבל מעט עגבניות בצורה רציפה במקום הרבה עגבניות בבת אחת, ובכך גם אורך החיים של השיח מתארך. בכדי לגרום לשיח העגבניות לצמוח כלפי מעלה, השיח מלוף סביב מוט מתכת גבוה שמטרתו לתמוך בשיח, ואת השיח חוסמים מסביב באמצעות כלוב בכדי שלא ייתפשט לרוחב, ניתן לראות דוגמא לכך באיור 6.9. מובן שכאשר השיח גדל מרחיבים את קוטר הגדר החוסמת את השיח.



איור 6.9 : גידול שיח עגבניות באמצעות כלוב.

כעת, באנלוגיה לבעיה שלנו, אם העץ הוא שיח העגבניות אזי נרצה לבסס אותו סביב עמוד מתכת אנלוגי תוך כדי שימוש בכלוב המתאים למימדיו. הרעיון הוא כדלהלן: נבנה את גרף וורונוי הכללי עבור מרחב העבודה הנתון. ע"י שימוש בגרף וורונוי הכללי שהתקבל ובאמצעות האלגוריתם של דייקסטרה נמצא את המסלול הקצר ביותר בגרף בין צומת המוצא למצב היעד הרצוי. באנלוגיה שלנו, גרף וורונוי הכללי יהיה עמוד המתכת עליו מתבסס השיח. לאחר מכן, הפונקציה אשר מייצרת מצבים רנדומליים תייצר מספר מוגדר של מצבים רנדומליים הנמצאים בתוך "כלוב" -

נפח מוגדר המיוצר באמצעות משיכה של מעגל בקוטר מוגדר בעקבות המסלול שנוצר על סמך גרף וורונוי הכללי. לצורך המחשה ניתן להתבונן באיור הבא:



איור 6.10: אופן יצירת ה"כלוב" באמצעות גרף וורונוי הכללי.

לאחר שנוצר מספר מצבים מוגדר, נגדיל את קוטר המעגל, נייצר מספר מצבים מוגדר וחוזר חלילה (הגדלת הקוטר וייצור מספר מצבים מוגדר). חשוב לציין כי רק מיקום מרכז המסה נבחר רנדומלית מתוך נפח הכלוב, ושאר הפרמטרים (אוריינטציה ומהירות) נבחרים רנדומלית מתוך התחום המוגדר עבור כל פרמטר.

באמצעות שיטת זו למעשה נשמור על מרבית היתרונות של אלגוריתם ה- RRT: אין צורך בבניית מרחב המצב או מרחב הקונפיגורציה, ישנה התחשבות באילוצי הדינמיקה כמו גם התחשבות בהגבלות על גודל הכניסות, ובנוסף המצבים הרנדומליים נוצרים עם התחשבות משמעותית בטופולוגיה של מרחב העבודה. יתרון נוסף שתיתן לנו שיטה זו הוא כי בעת קבלת המסלול מהאלגוריתם של דייקסטרה בחלק מהמקרים בהם לא ניתן לעמוד במטרה באופן חד-משמעי (לדוגמא אם יש קיר בין המוצא ליעד) האלגוריתם יוכל להודיע מראש כי אין פתרון ולחסוך זמן יקר. כמובן כי עדיין קיימים מצבי ביניים בהם לא נוכל לדעת באופן חד-משמעי כי אין פתרון למרות שזהו המצב, אך למרות זאת עבור מצבים ברורים האלגוריתם יידע לתת תשובה כי אין פתרון לעומת ה- RRT.

חסרון נוסף של האלגוריתם הוא כי יש לספק לו את גרף וורונוי הכללי. כפי שראינו בניית הגרף במרחב עבודה גדול עשויה להימשך כמספר דקות. למרות זאת, בניית גרף וורונוי הכללי נעשית פעם אחת עבור מרחב עבודה נתון, ובמקרה של שינוי המוצא או היעד נשתמש פעם נוספת בגרף הנ"ל ופשוט נחבר אליו את המוצא והיעד העדכניים. ה- RRT ו- RRTT כל שינוי של המוצא או היעד דורש חישוב חדש לגמרי. לכן, עבור מרחב עבודה לו ניתן לבנות בקלות את גרף וורונוי הכללי או עבור מערכת לה נרצה לחשב מספר תנועות ממצבים התחלתיים שונים למצבי יעד שונים ברור היתרון שבשימוש אלגוריתם ה- RRTT.

להלן הפסאודו-קוד עבור אלגוריתם ה- RRTT :

```

1  function RRTT ( Workspace, GVG,  $x_{start}$ ,  $x_{goal}$  ) :
2      Path = Dijkstra( GVG,  $x_{start}$ ,  $x_{goal}$  ) ;
3      if Path =  $\emptyset$  :
4          return No Solution ;
5       $T_{initial} = x_{start}$  ;
6      for  $k = 1$  to  $K$  :
7           $x_{random} = \text{Random\_State2}( Path, x_{goal}, k, K )$  ;
8          Extend(  $T$ ,  $x_{random}$  ) ;
9          if  $T$  reached goal :
10             Trajectory = go back from  $x_{goal}$  to  $x_{start}$  on  $T$  ;
11             return Trajectory ;
12      return  $T$  ;

```

איור 6.11 : פסאודו-קוד עבור אלגוריתם ה- RRTT.

כאשר הפונקציה Random_State2 נתונה ע"י הפסאודו-קוד הבא :

```

1  function Random_State2( Path,  $x_{goal}$ ,  $k$ ,  $K$ , Workspace ) :
2      static int  $n = 1$  ;
3       $D_{max}$  = greatest distance from any point on Path to Workspace bounds ;
4      if  $k \% \text{const1} = 0$  :
5          return  $x_{goal}$  ;
6      if  $k \% \text{int}(K/\text{const2}) = 0$  :
7           $n = n + 1$  ;
8       $D = (D_{max} / \text{const2}) \cdot n$  ;
9       $P$  = pipe with diameter  $D$  and guideline Path ;
10      $x_{random} = \text{Random\_State}(Workspace)$  ;
11      $x_{random}(1:3) = \text{Random\_Position}(P)$  ;
12     return  $x_{random}$  ;

```

איור 6.12 : הפסאודו-קוד עבור הפונקציה Random_State2.

כאשר $a \% b$ היא השארית הנוותרת מהחלוקה $a \div b$. הקבוע const1 מציין כל כמה איטרציות ייבחר מצב היעד כצומת הרנדומלי. הקבוע const2 מתאר לכמה צינורות עם קטרים שונים חילקנו את הניחושים שלנו. עבור const2 גדול נקבל צינור בעל קוטר גדול יותר כעבור מספר מועט של איטרציות, אך יהיו לנו הרבה מאוד צינורות כאלו, בעוד שעבור const2 קטן נקבל

מספר קטן של צינורות כאשר בכל צינור מספר רב של ניחושים. הפונקציה $Random_Position$ מייצרת ווקטור מיקום רנדומלי בתוך הצינור P .

6.3.2 הוכחת שלמות ההסתברותית

למעשה בעצם הוכחת שלמות ההסתברותית עבור אלגוריתם ה- RRT הוכחנו כי גם אלגוריתם ה- $RRTT$ שלם הסתברותית, וזאת משום שבסופו של דבר נגיע למצב בו האלגוריתם המשופר מייצר מצבים רנדומליים מתוך כלל מרחב המצב החופשי, משום שקוטר הכלוב יהיה גדול מספיק כך שהכלוב יכיל את כל המרחב. לכן בסוף מדובר למעשה ב- RRT , ולכן במידה ונמשיך לבצע עוד אינסוף איטרציות, גם באלגוריתם ה- $RRTT$ ההסתברות שנמצא את הפתרון במידה והוא קיים שואפת ל- 1. למעשה, ניתן לראות כי אלגוריתם ה- $RRTT$ שואף להפוך לאלגוריתם ה- RRT ככל שבוצעו יותר איטרציות.

6.3.3 הערכה לכמות האיטרציות הדרושה

ניזכר בהגדרות מתת-הפרק 6.2.3. השיפור המוצע לאלגוריתם ה- RRT נובע מהעובדה כי אנו חושבים כי רצף הפתרון יהיה על, או לפחות באיזור הקרוב של המסלול המיוצר באמצעות גרף וורונוי הכללי. עבור מרבית מרחבי העבודה הפשוטים בהם מספר המכשולים לא גדול מאוד, והמכשולים עצמם לא קטנים מאוד - הנחה זו כנראה תהיה נכונה.

לכן, במידה וההנחה נכונה, למעשה נקטין את הערך של $\mu(X_{free})$ משום שאנו מגבילים את הניחוש למיקום מרכז המסה להיות מהנפח הכלוא בכלוב. לכן, הערך של החסם התחתון להסתברות כי מצב רנדומלי ייפול בתוך תת-המרחב A_i (משוואה (154)) ייגדל, ומשום כך מספר האיטרציות הצפוי עד למציאת הפתרון ייקטן.

עבור מרחבי עבודה מרובי מכשולים ובעלי מכשולים קטנים לעומת זאת, למעשה משום שההנחה אינה נכונה, משמעות הדבר היא כי למעשה תת-המרחב A_i שוכן כולו, או חלק גדול ממנו מחוץ לכלוב. משום כך, למרות הקטנת הערך $\mu(X_{free})$ אנו בעצם מקטינים בצורה משמעותית מאוד את $\mu(A_i)$ ולעיתים אף מאפסים אותו. מובן כי בסופו של דבר נגיע למצב בו נפח הכלוב גדול מספיק והוא מכיל גם את רצף הפתרון, אך עבור מספר איטרציות רבות אנו עשויים לאלץ ייצור של מצבים ללא סיכוי להגעה לפתרון. כלומר מספר האיטרציות הצפוי יהיה גדול יותר - מספר האיטרציות הצפוי עבור ה- RRT המקורי יחד עם מספר האיטרציות שבוזבזו בהן נפח הכלוב לא הכיל את רצף הפתרון.

לסיכום, עבור מרבית מרחבי העבודה המעשיים, האלגוריתם המשופר צפוי לתת פתרון (במידה וקיים) בזמן מהיר יותר ועבור מרחבי עבודה מסובכים מאוד בהן נדרש תמרון מאוד מסובך האלגוריתם המשופר צפוי לתת פתרון בזמן איטי יותר. בנוסף, ברור כי עבור מצב בו אין פתרון בצורה חד-משמעית (כדוגמת קיר חוסם) האלגוריתם המשופר יימנע מביצוע האלגוריתם הרנדומלי וייתן תשובה מהירה כי לא קיים פתרון.

6.4 סימולציות לתכנון תנועה

נראה עתה סימולציות תכנון תנועה שבוצעו בשתי השיטות השונות עבור מספר מרחבי עבודה מוגדרים. לאחר מכן נבצע השוואה בין ביצועי האלגוריתמים השונים. פונקציית המרחק (Heuristic) בה השתמשנו היא מהצורה הבאה:

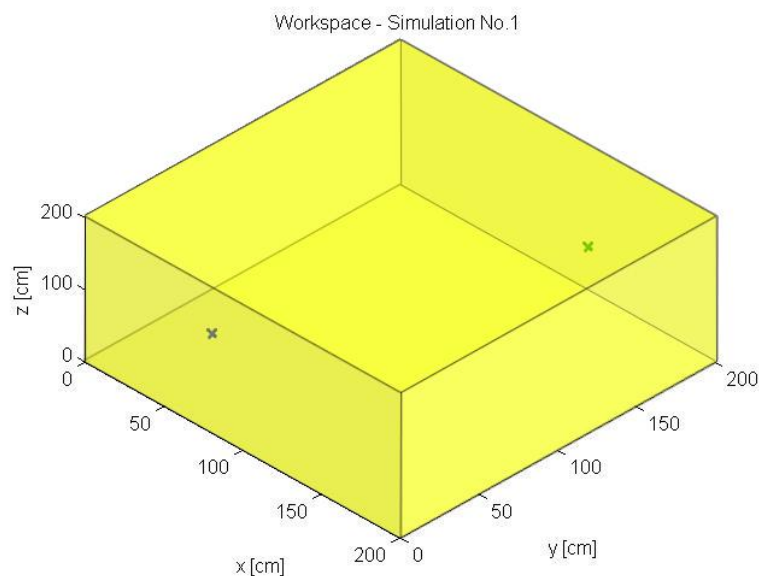
$$\rho(x_1, x_2) = w_{position} \|p_1 - p_2\| + w_{velocity} \|v_1 - v_2\| + w_{a.velocity} \|\Omega_1 - \Omega_2\| + \dots \quad (156)$$

$$\dots + w_{orientation} \|\text{trace}(\text{eye}(3) - R_1^T R_2)\|$$

כאשר p_1, p_2 הינם מיקום מרכז המסה של כלי הטיס מווקטורי המצב x_1, x_2 בהתאמה ו- $w_{position}, w_{a.velocity}, w_{velocity}, w_{orientation}$ הינם משקלים. ערכי המשקלים בהם השתמשנו הם כדלהלן: $w_{position} = 0.8, w_{a.velocity} = 0.1, w_{velocity} = 0.3, w_{orientation} = 0.3$. בנוסף, נציין כי פרק הזמן הנבחר עבור התקדמות נומרית בזמן בכל איטרציה היה $\Delta t = 0.25[\text{sec}]$.

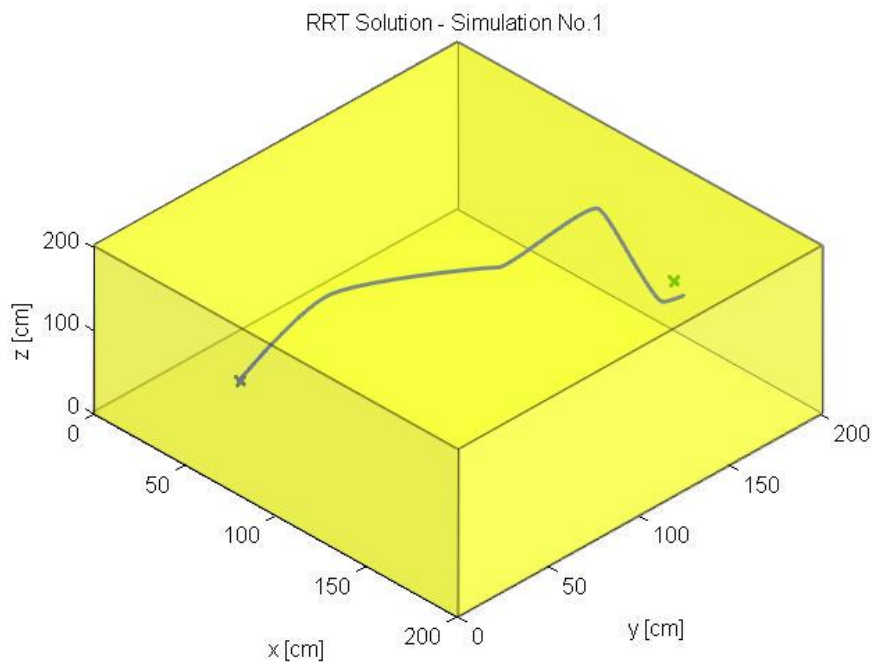
6.4.1 סימולציות לתכנון תנועה באמצעות אלגוריתם ה-RRT

תכנון התנועה הראשון בוצע עבור מרחב עבודה בגודל $200\text{cm} \times 200\text{cm} \times 200\text{cm}$, ללא מכשולים מלבד לדפנות מרחב העבודה. מצב המוצא שנבחר היה טיסה אופקית (ללא מהירות) סביב הנקודה $p_{start} = [40\text{cm}, 40\text{cm}, 40\text{cm}]$, ומצב היעד היה טיסה אופקית סביב הנקודה $p_{goal} = [160\text{cm}, 160\text{cm}, 160\text{cm}]$. נציין כי סימולציה זו בוצעה לצורך המחשה. את מרחב העבודה, מיקום המוצא ומיקום היעד ניתן לראות באיור הבא:



איור 6.13: מרחב העבודה עבור סימולציית תכנון התנועה הראשונה.

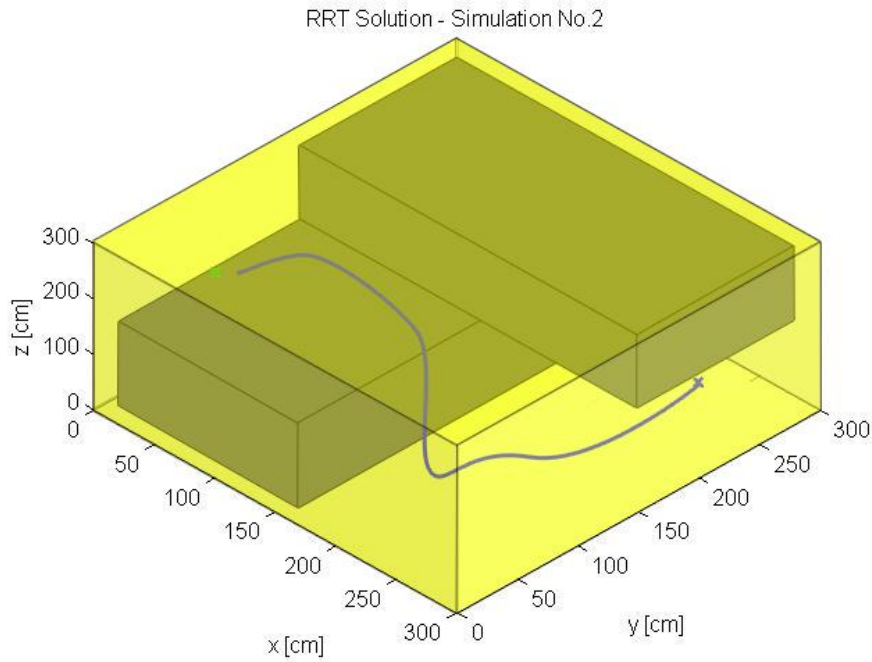
אלגוריתם ה- RRT הופעל עשר פעמים עבור מרחב עבודה זה. משך הזמן המינימלי הנדרש לתכנון התנועה היה $262.8096[\text{sec}]$ ונדרשו 2365 איטרציות בכדי לתכנן את התנועה. משך הזמן המקסימלי הנדרש לתכנון התנועה היה $507.3992[\text{sec}]$ ונדרשו 3896 איטרציות בכדי לתכנן את התנועה. משך הזמן הממוצע לתכנון התנועה עבור מרחב עבודה זה היה $405.8222[\text{sec}]$ ומספר האיטרציות הממוצע היה 3483. באיור 6.14 ניתן לראות את אחד הפתרונות שהתקבלו עבור מרחב עבודה זה. חשוב לזכור כי התכנון נעשה במרחב שנים-עשר מימדי בעוד אנו יכולים להציג מרחב תלת-מימדי לכל היותר.



איור 6.14 : אחד הפתרונות שהתקבלו באמצעות RRT עבור סימולציית התכנון הראשונה.

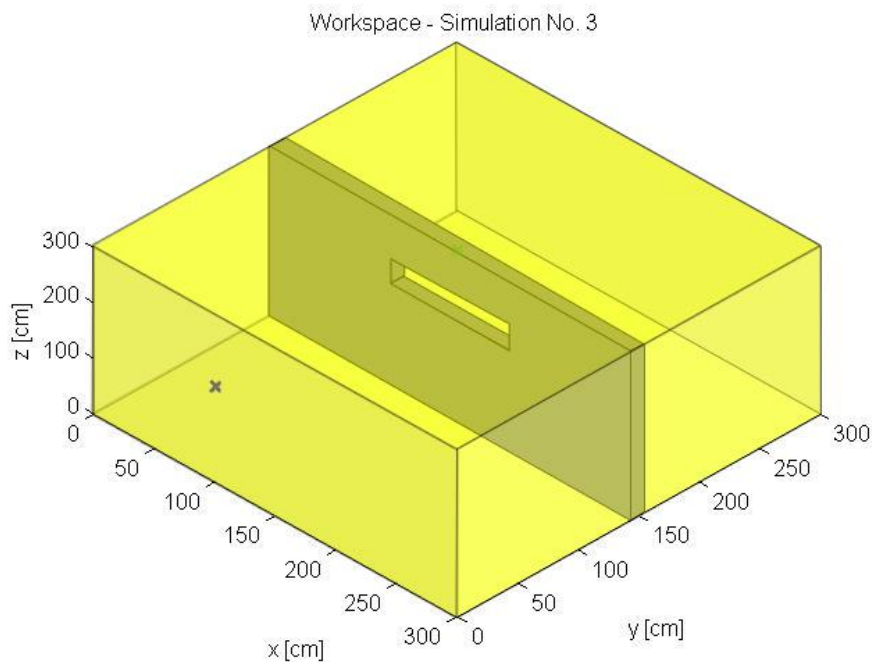
תכנון התנועה השני בוצע עבור מרחב העבודה המוצג באיור 5.15. מצב המוצא שנבחר היה טיסה אופקית (ללא מהירות) סביב הנקודה $p_{start} = [250\text{cm}, 250\text{cm}, 50\text{cm}]$, ומצב היעד היה טיסה אופקית סביב הנקודה $p_{goal} = [50\text{cm}, 50\text{cm}, 250\text{cm}]$.

אלגוריתם ה- RRT הופעל עשר פעמים עבור מרחב עבודה זה. משך הזמן המינימלי הנדרש לתכנון התנועה היה $299.6813[\text{sec}]$ ונדרשו 2532 איטרציות. משך הזמן המקסימלי הנדרש לתכנון התנועה היה $1634.0134[\text{sec}]$ ונדרשו לאלגוריתם 9656 איטרציות. משך הזמן הממוצע הנדרש לתכנון התנועה עבור מרחב עבודה זה היה $816.8067[\text{sec}]$ ומספר האיטרציות הממוצע היה 4840. באיור 6.14 ניתן לראות את אחד הפתרונות שהתקבלו עבור מרחב עבודה זה.



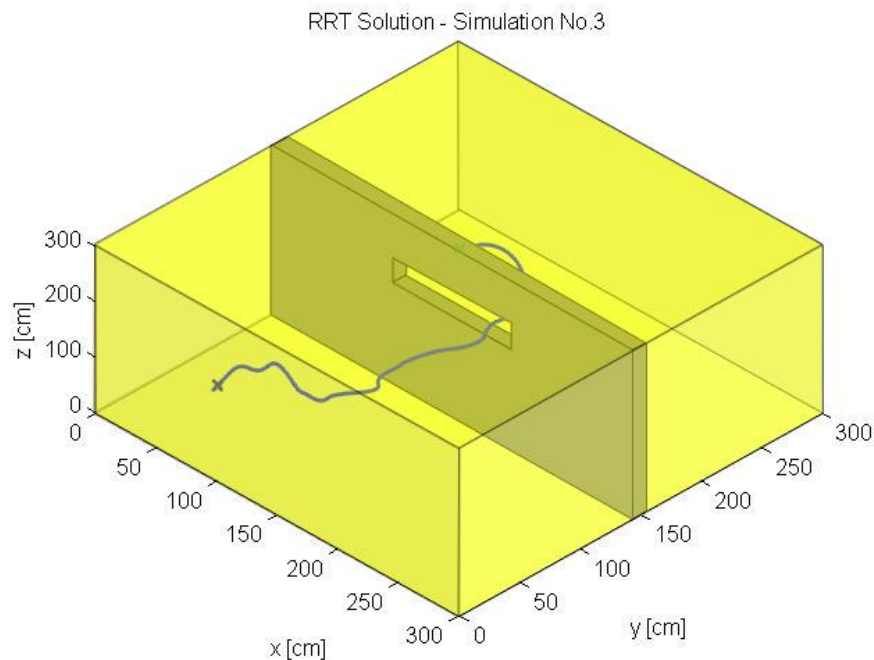
איור 6.15 : אחד הפתרונות שהתקבלו באמצעות RRT עבור סימולציית התכנון השנייה.

תכנון התנועה השלישי בוצע עבור מרחב עבודה בגודל $300cm \times 300cm \times 300cm$, בעל מכשול בצורת קיר עם פתח אופקי. מצב המוצא שנבחר היה טיסה אופקית (ללא מהירות) סביב הנקודה $p_{start} = [50cm, 50cm, 50cm]$, ומצב היעד היה טיסה אופקית סביב הנקודה $p_{goal} = [50cm, 250cm, 50cm]$. את מרחב העבודה, מיקום המוצא ומיקום היעד ניתן לראות באיור הבא :



איור 6.16 : מרחב העבודה עבור סימולציית תכנון התנועה השלישית.

אלגוריתם ה- RRT הופעל חמש פעמים עבור מרחב העבודה הנתון. משך הזמן המינימלי הנדרש לתכנון התנועה היה $7.3[hours]$ ומספר האיטרציות היה 58192. בשתיים מהפעמים שהופעל האלגוריתם, גם לאחר 100000 איטרציות לא הצליח האלגוריתם להגיע לפתרון. משך הזמן הממוצע (בהזנחת הכשלונות) למציאת פתרון היה $16.4[hours]$ ומספר האיטרציות הממוצע היה 71030. באיור הבא ניתן לראות את אחד הפתרונות שהתקבלו:

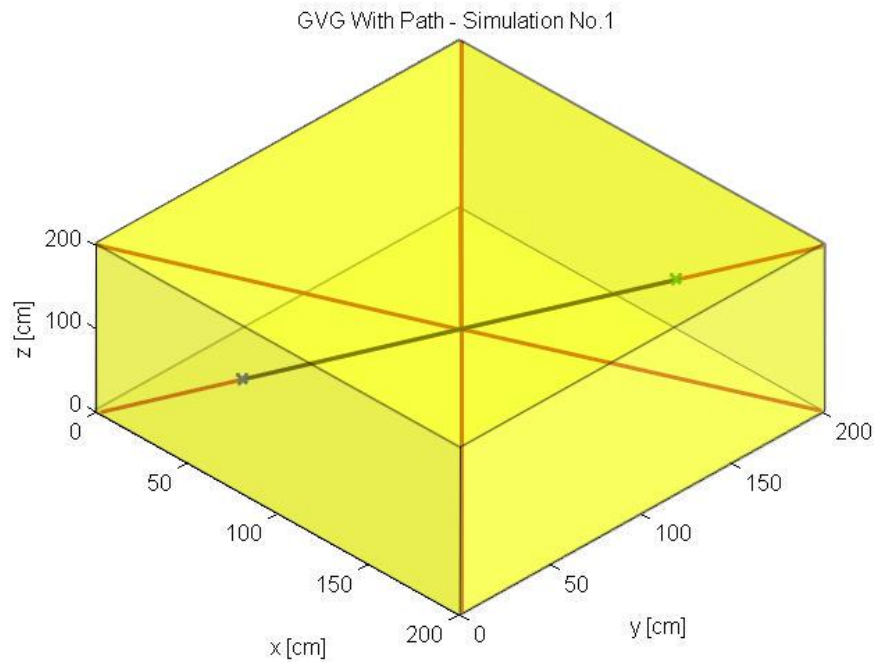


איור 6.17: אחד הפתרונות שהתקבלו באמצעות RRT עבור סימולציית התכנון השלישית.

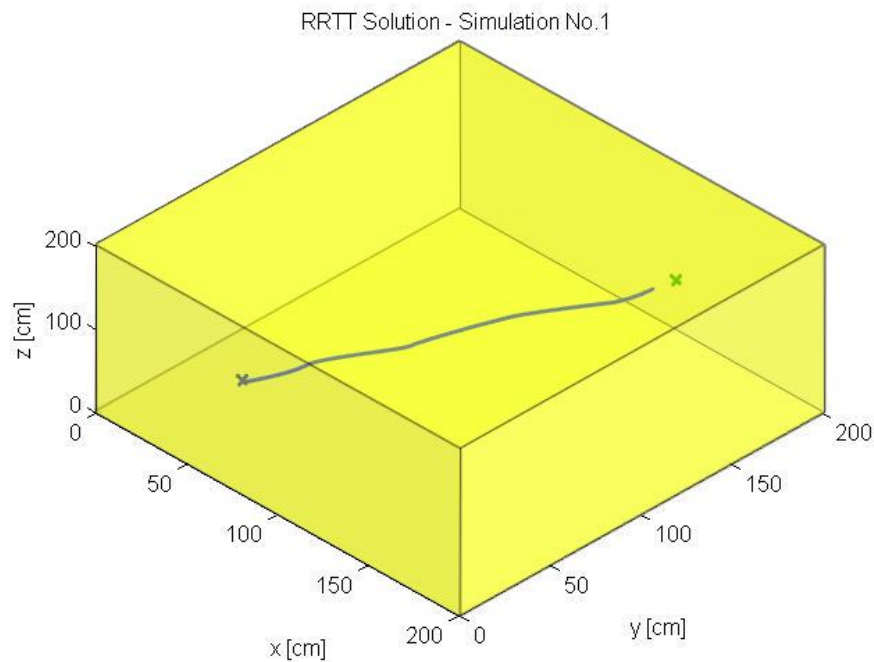
6.4.2 סימולציות לתכנון תנועה באמצעות אלגוריתם ה- RRTT

סימולציית התכנון הראשונה בוצעה עבור מרחב העבודה המוצג באיור 6.13. באיור 6.18 ניתן לראות את המסלול המתקבל לאחר שימוש באלגוריתם של דייקסטרא על גרף וורווי הכללי, כאשר באדום מסומן גרף וורווי הכללי ובשחור המסלול הקצר.

אלגוריתם ה- RRTT הופעל עשר פעמים עבור מרחב עבודה זה. משך הזמן המינימלי הנדרש לתכנון התנועה היה $57.3[sec]$ ונדרשו 827 איטרציות בכדי לתכנן את התנועה. משך הזמן המקסימלי הנדרש לתכנון התנועה היה $273.9702[sec]$ ונדרשו 2351 איטרציות בכדי לתכנן את התנועה. משך הזמן הממוצע לתכנון התנועה עבור מרחב עבודה זה היה $180.2226[sec]$ ומספר האיטרציות הממוצע היה 1901. באיור 6.19 ניתן לראות את אחד הפתרונות שהתקבלו עבור מרחב עבודה זה.



איור 6.18 : גרף וורונוי הכללי והמסלול הקצר עבור סימולציית התכנון הראשונה.

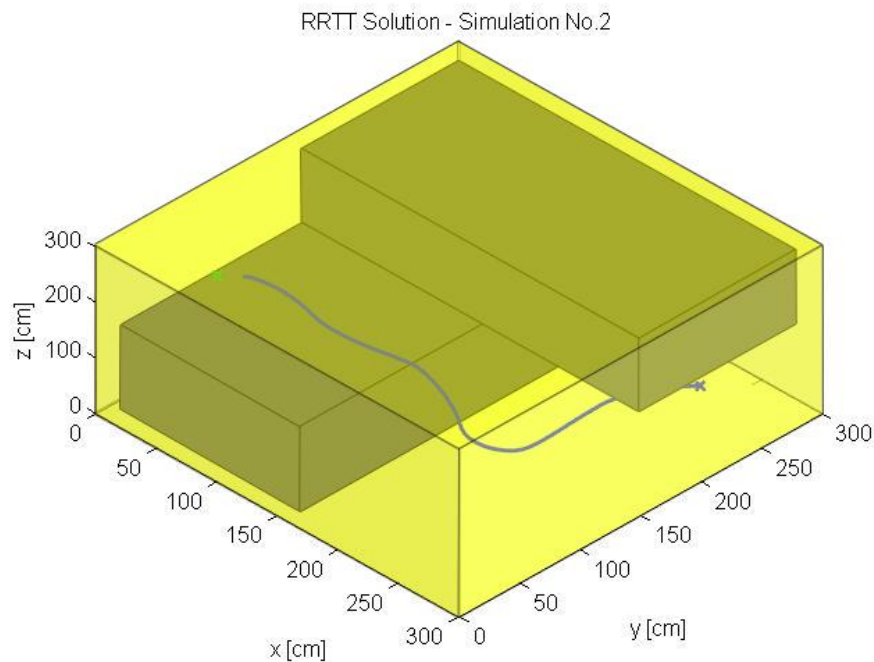


איור 6.19 : אחד הפתרונות שהתקבלו באמצעות RRTT עבור סימולציית התכנון הראשונה.

תכנון התנועה השני בוצע עבור מרחב העבודה המוצג באיור 5.15. באיור 5.22 ניתן לראות את המסלול הקצר המתקבל לאחר שימוש באלגוריתם של דייקסטרא על גרף וורונוי הכללי.

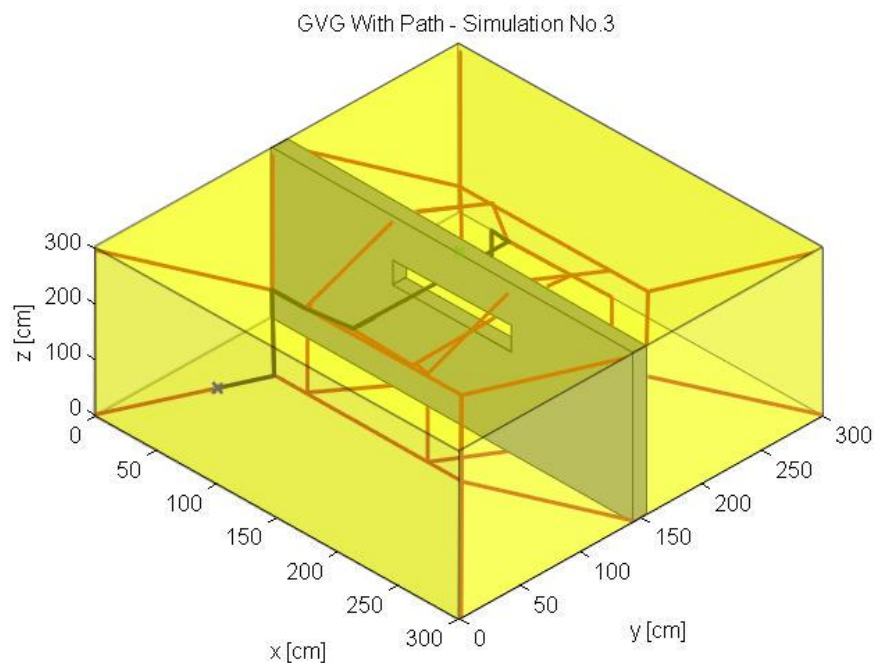
אלגוריתם ה- RRTT הופעל עשר פעמים עבור מרחב עבודה זה. משך הזמן המינימלי הנדרש לתכנון התנועה היה $78.1816[\text{sec}]$ ונדרשו 1030 איטרציות. משך הזמן המקסימלי הנדרש לתכנון התנועה היה $622.0011[\text{sec}]$ ונדרשו לאלגוריתם 6053 איטרציות. משך הזמן הממוצע

הנדרש לתכנון התנועה עבור מרחב עבודה זה היה $396.2496[\text{sec}]$ ומספר האיטרציות הממוצע היה 3615. באיור הבא ניתן לראות את אחד הפתרונות שהתקבלו עבור מרחב עבודה זה:



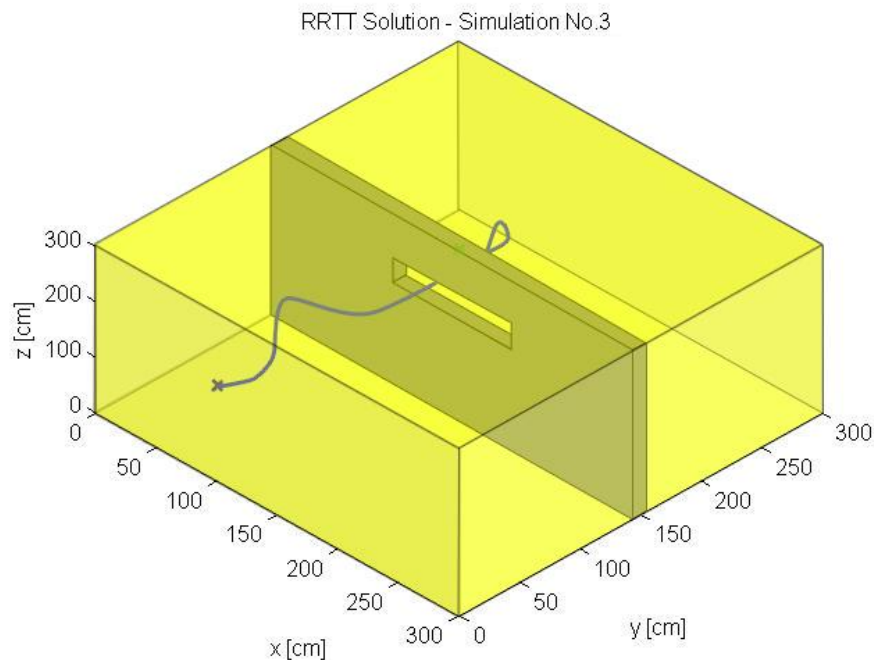
איור 6.20 : אחד הפתרונות שהתקבלו באמצעות RRT עבור סימולציית התכנון השנייה.

תכנון התנועה השלישי בוצע עבור מרחב העבודה המוצג באיור 6.16. באיור הבא ניתן לראות את המסלול הקצר המתקבל לאחר שימוש באלגוריתם של דייקסטרא על גרף וורונוי הכללי עבור מרחב עבודה זה:



איור 6.21 : גרף וורונוי הכללי והמסלול הקצר עבור סימולציית התכנון השלישית.

אלגוריתם ה- RRTT הופעל חמש פעמים עבור מרחב העבודה הנתון. משך הזמן המינימלי הנדרש לתכנון התנועה היה $1.09[hours]$ ומספר האיטרציות היה 10021. משך הזמן המקסימלי הנדרש לפתרון היה $25.873[hours]$ ומספר האיטרציות היה 98889. משך הזמן הממוצע למציאת פתרון היה $15[hours]$ ומספר האיטרציות הממוצע היה 67796. באיור הבא ניתן לראות את אחד הפתרונות שהתקבלו:



איור 6.22 : אחד הפתרונות שהתקבלו באמצעות RRTT עבור סימולציית התכנון השלישית.

6.4.3 סיכום תוצאות הסימולציות

מהסימולציות שבוצעו עבור שתי שיטות תכנון התנועה ניתן לראות כי עבור שני האלגוריתמים, ככל שישנם יותר מכשולים וכאשר קיימים מעברים צרים, משך הזמן הנדרש לפתרון הבעיה גדל משמעותית ביחס למרחבי עבודה פשוטים יותר. טבלה 6.1 מרכזת את משכי זמן הריצה ומספר האיטרציות הדרוש של שני האלגוריתמים עבור שלושת הסימולציות אשר בוצעו. לשם המחשה ניתן להתבונן בטבלה זו.

ניתן לראות שמשך הזמן הממוצע, כמו גם מספר האיטרציות הממוצע, קטן משמעותית בשימוש באלגוריתם ה- RRTT. בנוסף ראינו כי עבור מקרים בהם אלגוריתם ה- RRT לא מצליח לפתור את הבעיה, במיוחד מקרים בהם ישנו מעבר צר דרכו יש לעבור, ה- RRTT מצליח למצוא פתרון, אמנם בזמן ארוך יחסית, אך הוא מוצא פתרון.

Simulation Num.	Parameter	Min	Max	Avarage	Trials
1	RRT - time	262.8096[sec]	507.3992[sec]	405.8222[sec]	10
	RRTT - time	57.3[sec]	273.9702[sec]	180.2226[sec]	
	RRT - iterations	2,365	3,896	3,483	
	RRTT - iterations	827	2,351	1,901	
2	RRT - time	299.6813[sec]	1634.0134[sec]	816.8067[sec]	10
	RRTT - time	78.1816[sec]	622.0011[sec]	396.2496[sec]	
	RRT - iterations	2,532	9,656	4,840	
	RRTT - iterations	1,030	6,053	3,615	
3	RRT - time	7.3[hours]	No Solution	16.4[hours]	5
	RRTT - time	1.09[hours]	25.873[hours]	15[hours]	
	RRT - iterations	58,192	100,000	71,030	
	RRTT - iterations	10,021	98,889	67,796	

טבלה 6.1: השוואת משכי הריצה וכמות האיטרציות הדרושות עבור כל סימולציה.

כמו כן, מהתוצאות אפשר לראות כי בעוד שב RRT הפתרונות המתקבלים נראים מוזרים ולא הגיוניים ובנוסף מלאים ב"קפיצות", הפתרונות המתקבלים ב RRTT חלקים יותר ומתיישרים טוב יותר עם ההגיון. לדוגמא בסימולציה הראשונה - הפתרון המתקבל מן ה RRTT נע בתנועה הקרובה לקו ישר בין המוצא ליעד, לעומת זאת בפתרון המתקבל מה RRT נעשית תנועה של התרחקות מן היעד ורק לאחר מכן התקרבות אליו.

כמו כן ניתן לראות כי אורך המסלול המתבצע קצר יותר בתנועות שחושבו באמצעות אלגוריתם ה RRTT ביחס ל RRT. ניתן לראות את אורכי המסלולים הממוצעים עבור שלושת הסימולציות ועבור שני האלגוריתמים השונים בטבלה הבאה:

Algorithm	Avarage Path Length		
	Simulation 1	Simulation 2	Simulation 3
RRT	294.1945[cm]	756.4806[cm]	841.0061[cm]
RRTT	203.8883[cm]	467.6494[cm]	518.3389[cm]

טבלה 6.2: השוואת אורכי המסלולים הממוצעים המתקבלים.

בנוסף, ניתן לראות מן הסימולציות כי התנועה המחושבת לא מגיעה ממש עד לנקודת היעד עצמה וזאת משם שהוגדר מרחק ε אשר מגדיר הגעה. כל עוד המרחק בין שני מצבים קטן מן ε - זה, האלגוריתמים מתייחסים למצב זה כאל הגעה מהמצב הראשון לשני.

לסיכום, ניתן לראות כי במרחבי העבודה שנבדקו, אלגוריתם ה- RRTT משיג פתרונות חלקים והגיוניים יותר בזמן קצר יותר (בממוצע) מאשר אלגוריתם ה- RRT.

7 סיכום ומסקנות

מטרת עבודת מחקר זו הייתה בקרה ותכנון תנועה לכלי טיס בלתי מאויש מסוג קוואדרוטור. במהלך העבודה פותח המודל הדינמי המלא של הכלי ללא הזנחות או הנחות כלשהן, ולאחר מכן בוצעו הנחות הגיוניות שאיפשרו קבלה של משוואות תנועה איתן ניתן לעבוד ביתר קלות ונוחות אך עדיין מייצגות בצורה נכונה את הדינמיקה של הכלי. לאחר מכן הוצגה מערכת בקרה לא-ליניארית המאפשרת עקיבה הכוללת שלושה מצבי טיסה שונים, תוך הוכחות כי שלושת מצבי הטיסה מבצעים עקיבה כנדרש, ותוך ביצוע סימולציות להמחשה.

בשלב הבא הוצגה השיטה הקלאסית לתכנון מסלול, הכוללת בנייה של מרחב קונפיגורציה, שימוש במפת דרכים מסוג גרף וורוני הכללי ושימוש באלגוריתם של דייקסטרא למציאת המסלול הקצר ביותר בגרף - אלגוריתם חיפוש בגרף. כמו כן הוצגו סימולציות הממחישות הן את החשיבות של שימוש במרחב קונפיגורציה והן את הבעייתיות הכרוכה בכך.

לאחר מכן הוצג אלגוריתם ה- RRT לתכנון תנועה במרחב המצב, אשר ייתרונו הגדול הוא שהתכנון נעשה במרחב המצב ושניתן לקחת בחשבון את מהירות הכלי ואת הדינמיקה שלו, אך אין צורך לבנות את מרחב המצב. כמו כן, הוצגה הוכחת שלמות הסתברותית עבור השיטה ובוצע חישוב להערכת כמות האיטרציות הנדרש. בנוסף, הוסברו חסרונות ויתרונות האלגוריתם ובוצעו סימולציות עבורו.

לאחר הבנת אופן פעולת אלגוריתם ה- RRT, תוך הבנת יתרונותיו וחסרונותיו, הוצע אלגוריתם תכנון תנועה חדש השומר על מרבית היתרונות של אלגוריתם ה- RRT ובו זמנית מבטל כמה מחסרונותיו. אלגוריתם ה- RRTT שהוצע מתבסס על שימוש באלגוריתם ה- RRT, כאשר המצבים הרנדומליים הנוצרים, נבחרים בצורה חכמה ומכוונת יותר אשר מתבססת על גרף וורוני הכללי. הוכחנו כי אלגוריתם זה שלם הסתברותית ובמהלך חישוב להערכת כמות האיטרציות הנדרש הראנו כי עבור מרבית מרחבי העבודה הסטנדרטיים, האלגוריתם החדש יידרוש פחות איטרציות. לבסוף בוצעו סימולציות בכדי להראות את ביצועי האלגוריתם.

מהשוואה בין תוצאות הסימולציות ראינו כי אכן האלגוריתם שהוצע מאפשר תכנון תנועה מהיר יותר, וכמו כן הוא שלם יותר - במידה וברור כי לא קיים פתרון, האלגוריתם המשופר יידע להודיע כי אין פתרון, לעומת האלגוריתם המקורי אשר לא מסוגל לעשות זאת. כמו כן ראינו כי הפתרונות שמספק אלגוריתם ה- RRTT הינם חלקים יותר והגיוניים יותר מאשר הפתרונות אשר מספק אלגוריתם ה- RRT.

בנוסף, נבנה כלי הטיס והוקמה מערכת ניסוי בכדי לממש את תכנון התנועה הלכה למעשה. אמנם הניסוי נכשל, אך כישלון זה נובע ככל הנראה משגיאות במדידות ומהעובדה שיש למדל את המנועים בצורה טובה יותר. כישלון הנובע משגיאות מדידה וממודלים לא מדויקים מצביע על אפשרות לפיתוח אלגוריתם ה- RRT או ה- RRTT. עבודת המשך אפשרית עשויה להיות שילוב של אחד מאלגוריתמים אילו עם חישובי הסתברויות. לדוגמא אם נרצה לנוע מצומת נתון למצב

רנדומלי כלשהו נבצע סימולציה ונקבל "ענף" המייצג את תנועת כלי הטיס, אך ברור כי בעקבות השגיאות במודלים ובמדידות נקבל תנועה קרובה, אך שונה. שילוב השיטה עם חישובי הסתברויות ייתן לנו "קונוס הסתברות" במקום ענף שכזה, כך שנוכל לדעת בדוואות (בהסתברות גבוהה יותר) כי אם תוכנן מסלול, כלי הטיס יגיע ליעד למרות השגיאות הללו.

לסיכום, בוצעה עבודה רבה לבקרה של הקוואדרוטור, כמו גם לתכנון תנועה תוך הצגת שיפור לאלגוריתם קיים.

ביבליוגרפיה 8

- [1] Hassan K.Khalil, Nonlinear Systems Third Edition, Prentice Hall 2002.
- [2] Taeyoung Lee, Melvin Leok and N. Harris McClamroch, "Control of Complex Maneuvers for a Quadrotor UAV using Geometric Methods on $SE(3)$ ", Cornell University Library, <http://arxiv.org/abs/1003.2005>.
- [3] James J. Kuffner Jr., Steven M. Lavalley, "Randomized Kinodynamic Planning", ICRA '99, http://www.kuffner.org/james/papers/lavalle_icra1999.pdf.
- [4] Samir Bouabdallah, Andre Noth and Roland Siegwart, "PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor", Swiss Federal Institute of Technology.
- [5] Samir Bouabdallah, Pierpaolo Murrieri, Ronald Siegwart, "Design and Control of an Indoor Micro Quadrotor", 2004 IEEE International Conference on Robotics & Automation, New Orleans, LA - April 2004.
- [6] Geoffrey Vining (2011), Statistical Methods for Engineers, Cengage Learning.
- [7] Mu Huang, Bin Xian, Chen Diao, Kaiyan Yang and Yu Feng, "Adaptive Tracking Control of Underactuated Quadrotor Unmanned Aerial Vehicles via Backstepping", 2010 American Control Conference, Marriott Waterfront, Baltimore.
- [8] Richard M. Murray, Zexiang Li, S. Shankar Sastry, A Mathematical Introduction to Robotic Manipulation, CRC Press 1994.
- [9] Howie Choset (2005), Principles of Robot Motion, MIT.
- [10] Jean-Claude Latombe (1993), Robot Motion Planning, Kluwer Academic Publishers.
- [11] Ashfaq Ahmad Mian, Wang Daobo, "Nonlinear Flight Control Strategy for an Underactuated Quadrotor Aerial Robot", Nanjing University of Aeronautics and Astronautics.
- [12] Brian Armstrong, Oussama Khatib, Joel Burdick, "The Explicit Model and Inertial Parameters of the PUMA 560 Arm", IEEE 1986.
- [13] L. Derafa, T. Madani, A. Benallegue, "Dynamic Modelling and Experimental Identification of Four Rotors Helicopter Parameters", IEEE 2006.
- [14] Reuven Segev (1993), The Dynamics of Particles and Rigid Bodies, BGU.
- [15] Samir Boubdallah (2007), Design and Control of Quadrotors with Application to Autonomous Flying, PhD Thesis, Lausanne Polytechnic University.
- [16] Samir Boubdallah, Roland Siegwart, "Backstepping and Sliding-Mode Techniques Applied to an Indoor Micro Quadrotor", ICRA 2005.
- [17] S. Boubdallah, A.Noht, R.Siegwart, "PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor", Intelligent Robots and Systems 2004.

- [18] P. Castillo, R. Lozano, A. Dzul, "Stabilization of a Mini Rotorcraft With Four Rotors", IEEE 2005.
- [19] Howie Choset, Joel Burdick, "Sensor Based Planning, Part I: The Generalized Voronoi Graph", IEEE 1995.
- [20] Howie Choset, Joel Burdick, "Sensor-Based Exploration: The Hierarchical Generalized Voronoi Graph", The International Journal of Robotics Research, February 2000 vol. 19 no.2, 96-125.
- [21] Kenneth E. Hoff, Tim Culver, John Keyser, Ming Lin, Dinesh Manocha, "Fast Computation of Generalized Voronoi Diagrams Using Graphics Hardware", SIGGRAPH 1999, Proceedings of the 26th annual conference on Computer graphics and interactive techniques, 277-286.
- [22] Jongwoo Kim, James P. Ostrowski, "Motion Planning of Aerial Robot Using Rapidly-exploring Random Trees With Dynamic Constraints", IEEE 2003.
- [23] Hanna Kurniawati, Tirthankar Bandyopadhyay, "Global Motion Planning Under Uncertain Motion, Sensing, and Environment Map", Autonomous Robots, 2012 vol. 33 no. 3, 255-272.
- [24] S. Bhattacharya, M. Likhachev, V. Kumar, "Topological Constraints in Search-Based Robot Path Planning", Autonomous Robots, 2012 vol.33 no.3, 273-290.
- [25] A.A. Mian, W. Daobo, "Nonlinear Flight Control Strategy for an Underactuated quadrotor aerialrobot", IEEE 2008.
- [26] A.A Mian, W. Daobo, "Dynamic Modeling and Nonlinear Control Strategy for an Underactuated Quad Rotor Rotorcraft", Journal of Zhejiang University-Science, 2008 vol.9 no.4, 539-545.
- [27] James J. Kuffner, Steven M. LaValle, "RRT-Connect: An Efficient Approach to Single Query Path Planning", IEEE 2000.
- [29] C. Goerzen, Z. Kong, B. Mettler, "A Survey of Motion Planning Algorithms from the Perspective of Autonomous UAV guidance", Journal of Intelligent & Robotic Systems, 2010 vol. 57 no. 1, 65-100.
- [30] M. Hwangbo, J. Kuffner, T. Kanade, "Efficient Two-Phase 3D Motion Planning for Small Fixed-Wing UAVs", IEEE 2007.
- [31] Y. Kuwata, J. How, "Three Dimensional Receding Horizon Control for UAVs", AIAA 2004.
- [32] Taeyoung Lee, "Robust Adaptive Geometric Tracking Controls on $SO(3)$ With an Application to the Attitude Dynamics of a Quadrotor UAV", Cornell University Library, <http://arxiv.org/pdf/1108.6031v1.pdf>.
- [33] P.J Bristeau, P. Martin, E. Salaun, N. Petit, "The Role of Propellor Aerodynamics in the Model of a Quadrotor UAV", ECC 2009.

- [34] P. Pounds, R. Mahony, J.Gresham, P.Corke, J. Roberts, "Towards Dynamically-Favourable Quad-rotor Aerial Robot", Australian Robotics & Automation Association 2004.
- [35] Daniel Mellinger, Nathan Michael, Vijay Kumar, "Trajectory Generation and Control for Precise Aggressive Maneuvers with Quadrotor", IJRR April 2012 vol. 31 no. 5, 664-674.
- [36] Gabriel M. Hoffman, Steven L. Waslander, "Quadrotor Helicopter Trajectory Tracking Control", AIAA Navigation and Control Conference and Exhibit, Honolulu, 2008.
- [37] Patrick M. Bouffard, Steven L. Waslander, "A Hybrid Randomized/Nonlinear Programming Technique for Small Aerial Vehicle Trajectory Planning in 3D", IROS 2009.

נספח א' - ניסויים

במהלך עבודת מחקר זו, מלבד עבודה תיאורטית רבה נעשתה גם עבודה מעשית בבניית הקוואדרוטור ובהקמת מערכת ניסוי בכדי לאפשר ביצוע ניסויים. כמו כן, נעשה ניסיון לבצע ניסוי בו כלי הטיס יעבור ממצב נתון למצב רצוי על סמך תכנון התנועה המקדים.

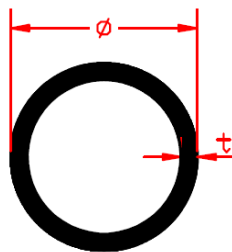
בפרק זה נתאר בקצרה את תהליך הבנייה של הכלי ואת הרכיבים בהם נעשה שימוש. כמו כן נסביר על מערכת הניסוי שהוקמה ונסביר על אופן אפיון מערכת הניסוי (זיהוי פרמטרים). לסיום נציג את תוצאת הניסיון לביצוע הניסוי.

1.א בניית כלי הטיס

נסביר עתה על תהליך בניית הכלי מהרמה של בניית גוף הכלי עצמו, תוך הצגת הרכיבים האלקטרוניים בהן השתמשנו ואופן הרכבתם, ועד לביצוע הכלי שנבנה.

1.1.א בניית גוף הכלי

תחילה, בכדי לבנות את הכלי יש לבנות את גוף הכלי - הצלב המשמש כשילדה של הכלי, בצורה כזו המאפשרת הרכבה של הרכיבים האלקטרוניים עליו כמו גם המנועים. הצלב נבנה ממוטות אלומיניום בקוטר של $\varnothing = 10mm$ ועובי של $t = 1mm$, ניתן לראות את פרופיל המוטות באיור הבא:



איור 1.א: פרופיל מוטות האלומיניום אשר שימשו לבניית שילדת הכלי.

הצלב נבנה כך שהמרחק מקצה אחד של המוט לקצה השני יהיה $60cm$ (כלומר המרחק ממרכז המסה למנוע יהיה $l = 30cm$) וזאת מפני שהכלי לא יהיה קטן מדי והפעלת כוח קטן ממנוע נתון תגרום לתזוזה משמעותית, ובנוסף בכדי שהכלי לא יהיה גדול מדי והפעלת כוח גדול ממנוע נתון לא תשפיע בצורה מספיקה על תנועת הכלי. נציין כי זוהי המידה המומלצת באתר של DIY Drones ע"י אנשים בעלי ניסיון. בכדי לבנות את הצלב, חצינו מוט אחד ושייפנו את דפנותיו כך שיתקבלו שקעים עגולים בקצוות בכדי שנוכל "להלביש" את החצאי המוטות על המוט השלם. לאחר מכן הדבקנו באמצעות דבק אפוקסי רב את המוטות. בנוסף, הדבקנו על החלק העליון פלטת פלסטיק ריבועית בכדי שבשלב מאוחר יותר נוכל להדביק אליה מושב לבטריה ואת הרכיבים האלקטרוניים.

לאחר בניית הצלב נאלצנו להכין תושבות בכדי לחבר את המנועים לקצות המוטות. בעוד הפרופיל של המוטות הוא עגול, התושבות של המנועים הן ריבועיות ולכן יש צורך במתאם בין תושבת המנוע למוטות. בכדי להכין מתאם זה לקחנו מוט אלומיניום בצורת U בעל מימדים מתאימים. חתכנו מוט זה לארבעה חלקים שווים ושייפנו את החלק הפנימי. בנוסף, שייפנו את החלק החיצוני של המוטות העגולים בכדי לקבל פני שטח ישרים עליהם נוכל להרכיב את המתאמים. את המתאמים חיברנו באמצעות דבק אפוקסי וברגים. בתמונה הבאה ניתן לראות את גוף הכלי שנבנה:



איור א.2: תמונה של גוף הכלי הבנוי.

בנוסף לכך, הוספה תושבת שתחזיק את הבטריה בזמן הטיסה. התושבת היא בצורת שולחן הפוך אשר חוברה באמצעות אפוקסי לתחתית הכלי. את הבטריה נחבר לתושבת באמצעות סקוויים.

א.2.1 הרכיבים האלקטרוניים והרכבתם

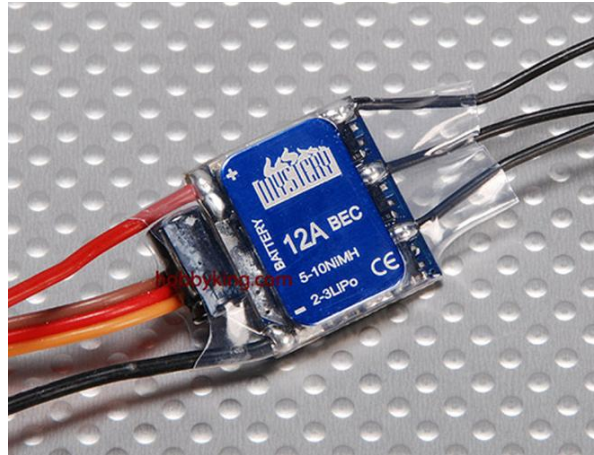
עתה, נפרט מהם הרכיבים האלקטרוניים בהם השתמשנו במהלך בניית הכלי. רכיבים אילו כוללים מנועים, מיקרו-בקרים, בקרי מהירות, פרופלורים, חישן אולטרה-סוני Xbee.

המנועים בהם השתמשנו הם המנועים: AX 2306N 1300kv brushless Micro Motor. תמונה לדוגמא של מנוע כזה ניתן לראות באיור הבא:



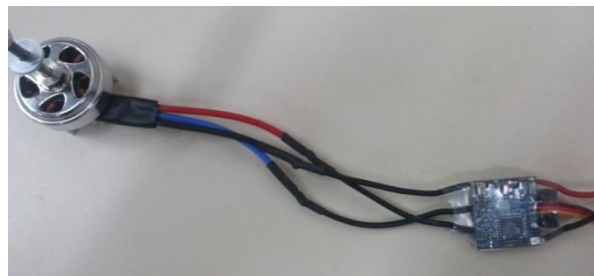
איור א.3: תמונה של מנוע Brushless Micro Motor.

בנוסף למנועים ישנו צורך להשתמש בבקרי מהירות אלקטרוניים (Electronic Speed Controller) שאחראים על סיבוב המנוע ע"י תזמון וחלוקת הזרמים המועברים למנוע על סמך המתח אשר מועבר לבקרים. את בקרי המהירות בהם השתמשנו ניתן לראות באיור הבא:



איור א.4: תמונה של בקרי המהירות בהם השתמשנו.

את שלושת החוטים שיוצאים מבקר המהירות הלחמנו לשלושת החוטים היוצאים מהמנוע בכדי לקבל את התוצאה הבאה:



איור א.5: תמונה של אופן חיבור בקר המהירות האלקטרוני למנוע.

בנוסף, חיברנו חיישן אולטרה-סוני בכדי לאפשר אחיזת גובה בזמן טיסה ידנית. החיישן שולח גלי קול, ולפי הזמן שלוקח לגלי הקול לחזור ניתן לדעת מה המרחק מהרצפה. החיישן בו השתמשנו הוא מסוג: Maxbotix LV-EZ4, אותו ניתן לראות באיור הבא:



איור א.6: החיישן האולטרה-סוני אשר שימש כחיישן גובה.

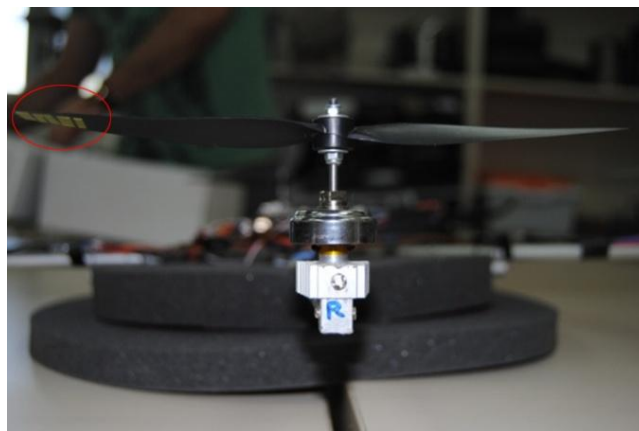
הרכיב היקר, הרגיש והחשוב ביותר הוא המיקרו-בקר (הכרטיס) עליו מתבצעים החישובים, ואשר מספק את המתחים לבקרי המהירות בהתאם לחישובים אילו. הכרטיס בו השתמשנו הוא למעשה שילוב של שני כרטיסים המחוברים האחד לשני. הכרטיסים הללו הם: ArduPilotMega IMU Shield ו- Arduino Compatible UAV Controller w/ATMega1280. באיור הבא ניתן לראות את שני הכרטיסים כאשר הם מחוברים:



איור א.7: תמונה של המיקרו-בקרים המחוברים האחד לשני.

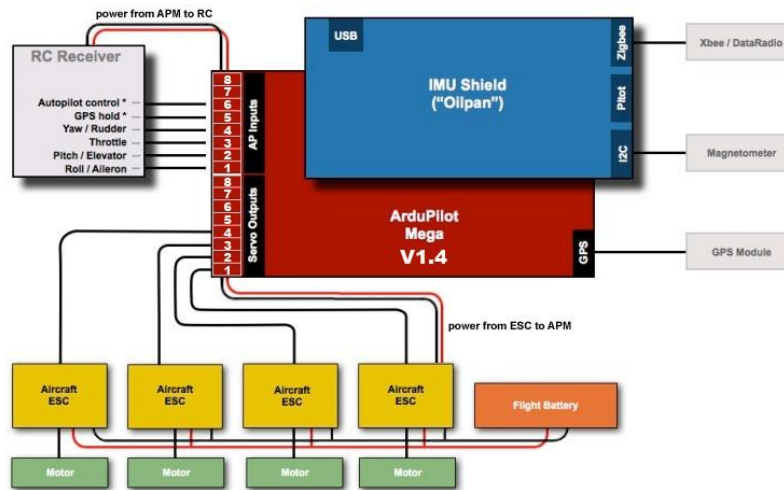
את הכרטיס חיברנו לגוף הכלי בצורה כזו שמד-התאוצה בכיוון z יהיה קרוב למרכז הכלי מפני שבצורה זו יתקבלו שגיאות מינימליות מה IMU (Inertial Measurement Unit). מלבד ביצוע כל הבקרה והפעלת מנועי המהירות, למיקרו-בקרים אילו מחוברים מדידי תאוצה, מצפן, מגנטומטר וגייירו (Gyros) אשר באמצעותם ניתן לחשב את אוריינטציית הכלי במרחב ואת המהירות הזוויתית של הכלי במרחב.

כעת, נחבר את הפרופלורים למנועים. לרוב, בעת הזמנה של פרופלורים זולים, הפרופלורים מגיעים לא מאוזנים - כלומר צד אחד של הפרופלור קל יותר מצידו השני וכתוצאה מכך מומנט האינרציה סביב ציר הסיבוב לא מתאפס, וסיבוב הפרופלור אינו מאוזן. בכדי לפתור בעיה זו אוזנו הפרופלורים ע"י הדבקת פיסות קטנות של דבק לצד הקל של הפרופלור עד לקבלת פרופלור מאוזן. דוגמא לפרופלור מאוזן ניתן לראות באיור הבא:



איור א.8: תמונה של פרופלור אשר אוזן באמצעות הדבקת פיסות דבק.

כעת, כל שנותר לעשות הוא לחבר את הרכיבים האלקטרוניים ע"פ סכימת החיבור הבאה:



איור א.9: סכימת חיבור כלל הרכיבים האלקטרוניים.

לרוב נהוג להוסיף רגליים שיהיו מספיק קשיחות בכדי להחזיק את הכלי ומצד שני שייספגו את המכה מבלי להישבר במהלך נחיתה או התרסקות. במקום להשתמש ברגליים הוספו ספוגים עגולים המכסים שטח נרחב מגוף הכלי, וזאת בכדי שבמקרה של נחיתה קשה, אפילו נחיתה בזווית, הספוג ייספוג את מרבית המכה ולא הכלי. באיור הבא ניתן לראות את הכלי הסופי שהתקבל במהלך טיסה באוויר:



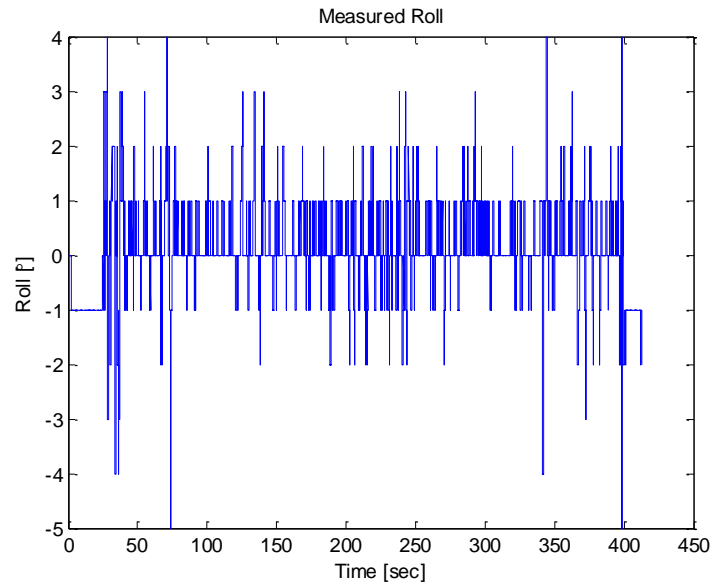
איור א.10: תמונה של כלי הטיס הסופי שהתקבל במהלך טיסה באוויר.

נציין בנוסף כי לכלי הוספנו XBee (זיגבי) בתדר $2.4GHz$ המאפשר לנו לבצע תקשורת אלחוטית בין המערכת שעל הכלי לבין מחשב PC לצורך ביצוע הניסוי.

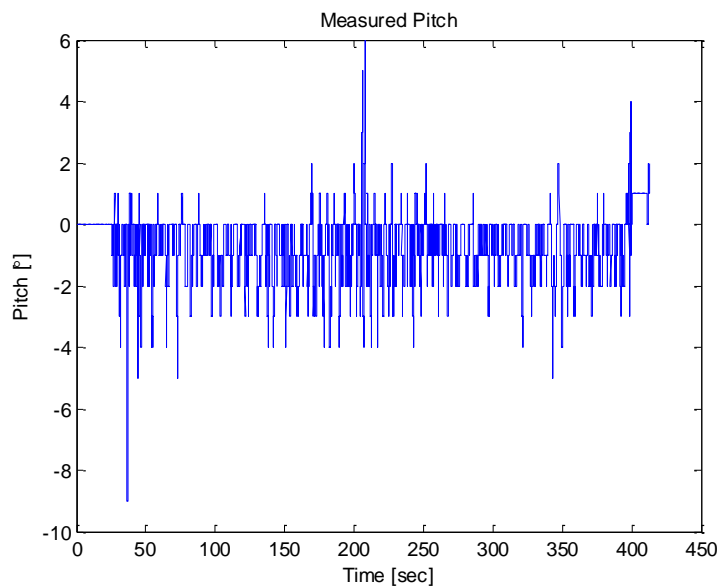
א.1.3 ביצועי הדגם

כפי שצינו, ישנם חיישנים הנמצאים על הכרטיסים באמצעותם ניתן לדעת מהי האוריינטציה הרצויה ביחס לעולם. הבקרה הממומשת על הכלי בעת טיסה רגילה באמצעות שלט רחוק (רדיו) היא בקרת PID. עבור כל זווית מבין שלושת הזוויות המתארות אוריינטציה (RPY) אנו משתמשים בבקרת PID כאשר הזווית הרצויה נשלחת לכלי באמצעות שלט.

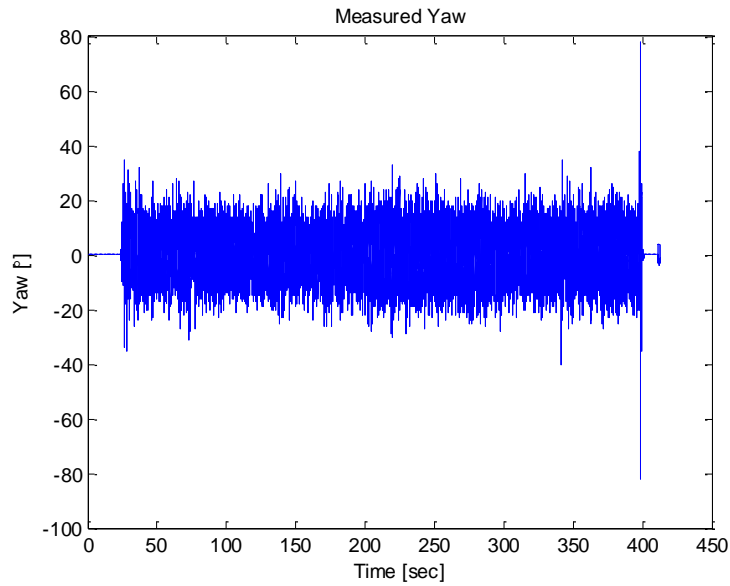
נציג עתה את התוצאות המתקבלות עבור עקיבה אחרי אות רצוי הנשלח משלט רחוק. האות אחריו אנו רוצים לעקוב נשלח ע"י רוחב פולס משתנה, אשר מיוצג ב- μs . שלושת הגרפים הבאים מציגים את הזוויות (RPY) אשר נמדדו מהכרטיס שעל הכלי בזמן טיסה. במהלך טיסה זו השליטה הייתה ידנית (כאמור, באמצעות שלט) ומטרתה הייתה ייצוב הכלי באוויר סביב נקודה כלשהי. להלן הגרפים:



גרף א.1: זווית ה-Roll אשר נמדדה בזמן טיסה ידנית.



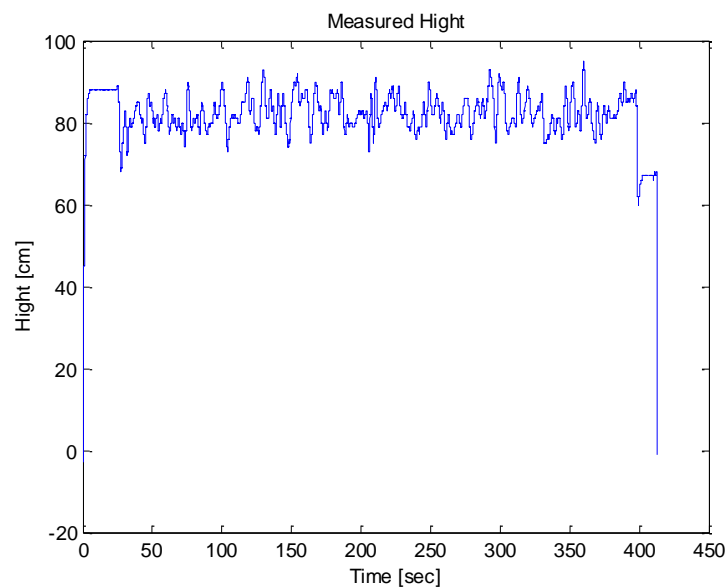
גרף א.2: זווית ה-Pitch אשר נמדדה בזמן טיסה ידנית.



גרף א.3 : זווית ה - Yaw אשר נמדדה בזמן טיסה ידנית.

ניתן לראות כי אם נסיר את רעש המדידה נקבל למעשה ממוצע הנע סביב האפס עבור שלושת הזוויות, מלבד הנקודות בהן יש שינויים קיצוניים. שינוי קיצוני שכזה מעיד על זווית שונה מאפס ולא על רעש מדידה.

בנוסף, כפי שצויין, ניתן באמצעות חיישן אולטרה-סוני לקבל אחיזת גובה. באמצעות שינוי מצב של אחד ממתגי השלט, הכלי משנה מצב טיסה ועובר למצב של אחיזת גובה, כאשר הגובה הרצוי הוא הגובה הנמדד בעת שינוי מצב המתג. בגרף הבא ניתן לראות את הגובה הנמדד (המרחק מן הרצפה) באמצעות חיישן אולטרה-סוני הממוקם בתחתית הכלי :



גרף א.4 : הגובה הנמדד בזמן טיסה במצב אחיזת גובה.

ניתן לראות כי מרגע המעבר למצב של אחיזת גובה, הגובה נשמר בקירוב עד לסיום הטיסה. נציין כי גם במקרה זה קיימים רעשי מדידה המסוננים באמצעות פילטרים שונים, אך עדיין קיימים. אם נסיר את הרעש נראה כי גובהו של הכלי כמעט ואינו משתנה.

לסיכום, ניתן לראות שביצועי הכלי בעת טיסה ידנית מספקים וטובים, משמע הכלי נבנה ופועל כהלכה.

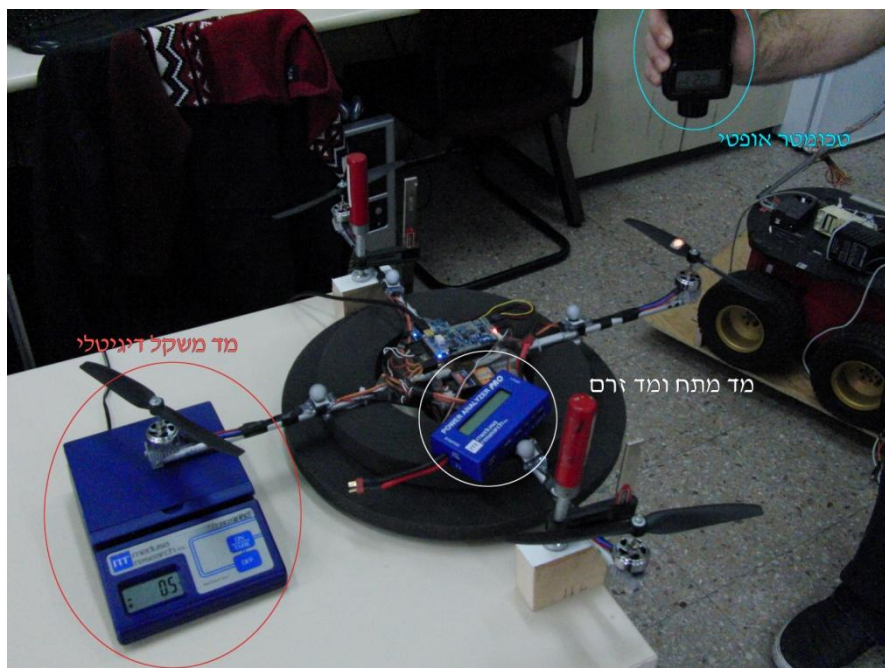
2. א. זיהוי פרמטרים עבור כלי הטיס

בכדי להשתמש בבקר שהוצג ובשיטות השונות לתכנון תנועה יש למצוא תחילה את כלל הפרמטרים המאפיינים את כלי הטיס שנבנה, פרמטרים כגון מומנטי אינרציה, מסה, מימדים ומקדמי העילוי והגרר עבור המנועים. בכדי לקבל פרמטרים אילו בוצעו מספר ניסויים אותם נציג עתה.

2.1. א. מציאת קבועי העילוי והגרר

בכדי להשתמש במערכת הבקרה שהצגנו, אנו מוכרחים לדעת מהו הקשר בין מהירות סיבוב המנוע, ובין אות הפיקוד הנשלח מן הכרטיס לבקר המהירות, לבין כוח העילוי שמפעיל כל פרופלור ולבין כוח הגרר הפועל על כל פרופלור. קשר זה הכרחי לשימוש בבקר משום שהבקר נותן לנו את גודל הכוחות והמומנטים אותם נרצה להפעיל, ובכדי להפעיל אותם בפועל יש לדעת מהו אות הפיקוד שיש לשלוח לכל מנוע ומנוע.

אם ניזכר במשוואות (1) ו- (5) נראה כי בכדי למצוא קשר זה יש למצוא את מקדם העילוי b ואת מקדם הגרר d . בכדי למדוד את מקדם העילוי b נבנתה מערכת הניסוי הבאה:



איור 11. א: תמונה של מערכת הניסוי למדידת קבוע העילוי.

הקוואדרטור הונח על בלוקי עץ קטנים ונעל באמצעות מהדקים (קלמרות) בכדי למנוע את תנועת הכלי ולאפשר לו לבצע תנועת נדנדה בלבד סביב מרכז המסה. בצורה זו, ע"י הפעלת המנוע החופשי למעשה נגרום למנוע הנגדי לו להפעיל כוח זהה כלפי מטה, כך שיידחף את מד-המשקל הדיגיטלי. ע"י קריאת משקל זו נוכל לקבל את הכוח הפועל. חשוב לציין כי המנוע החופשי נמצא במרחק של כמטר מהרצפה, כך שהוא חופשי מאפקטי קרקע העשויים לפעול עליו בעת קירבה לקרקע ולעוות את התוצאות.

ע"י הפעלת המנוע החופשי במספר אותות פיקוד קבועים ומדידה של הכוח המופעל באמצעות המשקל, מדידת אותות הפיקוד המועברים באמצעות המחשב (מחובר באמצעות כבל USB לכרטיס) ומדידת מהירות הסיבוב של המנוע באמצעות טכומטר אופטי נקבל את הקשרים בהם אנו מעוניינים. בנוסף, נמדדו הזרם והמתח המסופקים לבקר המהירות בכדי לקבל את ההספק הנצרך.

באופן דומה נבנתה מערכת ניסוי למדידת המומנט המופעל על כל מנוע בעת סיבוב. ע"י נעילת גובהו של המנוע אשר נמצא מנגד למנוע החופשי באמצעות בלוק עץ תחתיו, והעמדת מד-המשקל הדיגיטלי בצורה אנכית נוכל למדוד את הכוח האופקי הפועל על הכלי בעת הפעלת המנוע החופשי עבור מספר אותות פיקוד ($u_{command}$) שונים. ע"י מדידת הכוח האופקי והכפלתו באורך הזרוע בין המנוע לבין מרכז המסה של הכלי נוכל לקבל את המומנט הפועל על הפרופלור עבור אותות פיקוד שונים.

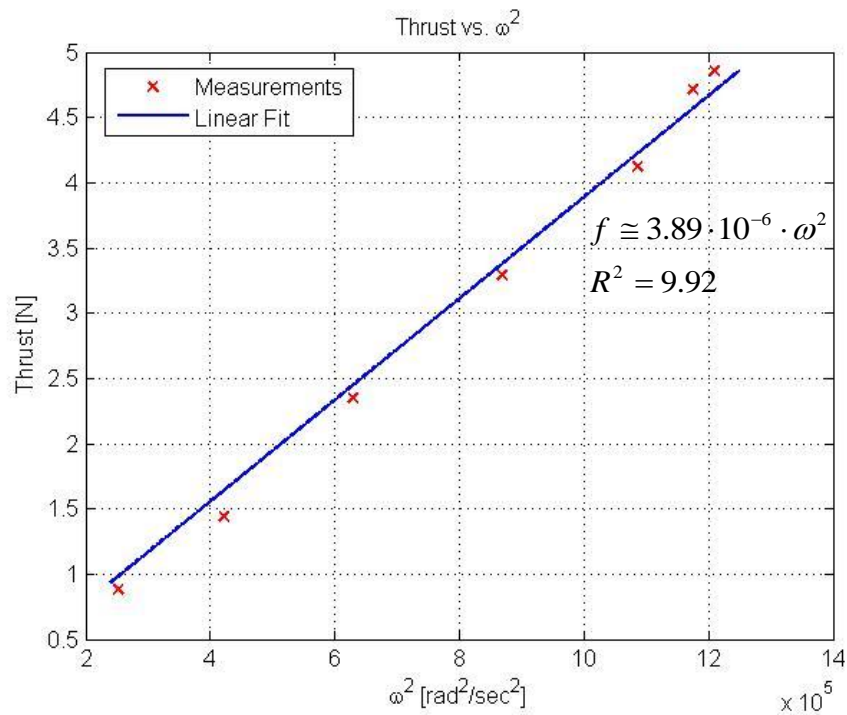
את תוצאות המדידות שהתקבלו בניסוי אפיון זה ניתן לראות בטבלה הבאה:

זרם [A]	מתח [V]	ω^2 [rad ² /sec ²]	ω [rad/sec]	ω [rpm]	מומנט [N · m]	דחף [N]	פיקוד [μsec]
0.83	12.28	252662	503	4800	0.49	0.88	1300
1.67	12.16	421542	649	6200	0.83	1.44	1400
2.80	12.00	629249	793	7575	1.25	2.35	1500
4.40	11.70	868635	932	8900	1.74	3.29	1600
7.00	11.40	1085684	1042	9950	2.17	4.12	1700
8.16	11.20	1174730	1084	10350	2.36	4.71	1800
8.50	11.55	1209027	1100	10500	2.44	4.86	1900

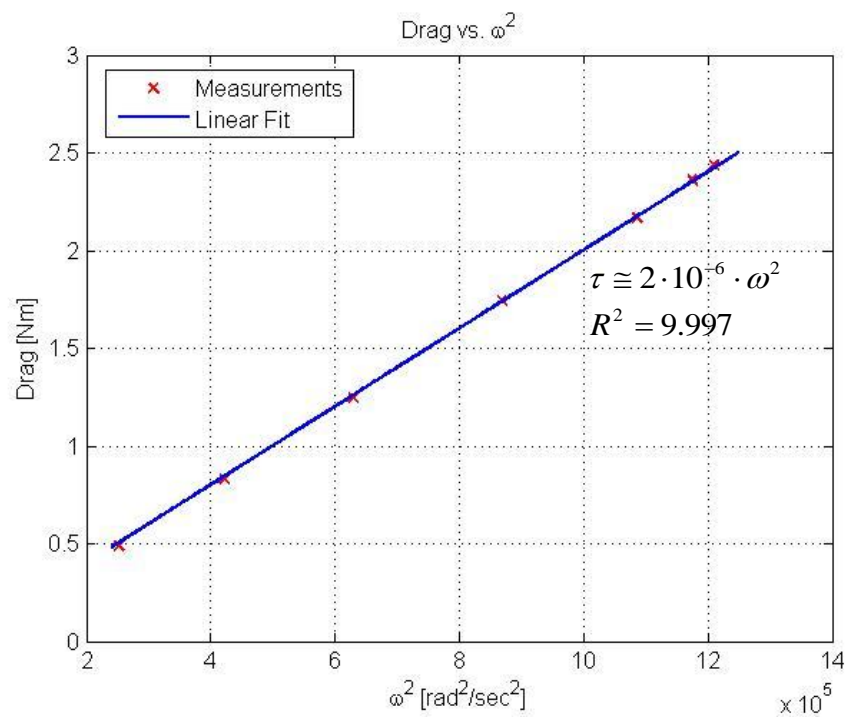
טבלה א.1: תוצאות המדידות שהתקבלו בניסוי למציאת קבוע העילוי וקבוע הגרר.

כאמור, אם נזכר במשוואות (1) ו- (5) נראה כי הקשר בין קבוע הגרר ובין קבוע העילוי למהירות סיבוב המנוע בריבוע הוא קשר ליניארי. ע"י העלאת הנתונים מן הטבלה הנ"ל על גרפים והעברת קווי מגמה ליניאריים נוכל לקבל את הערכים עבור קבוע העילוי ועבור קבוע הגרר. קשרים חשובים אפילו יותר מן הקשרים הללו הם הקשרים שבין אות הפיקוד הנשלח לבין כוח העילוי ולבין כוח הגרר. מסתבר כי קשרים אילו הינם ליניאריים גם כן, וע"י העלאת נתונים אילו על גרף גם כן, נוכל לקבל את הקשר בין הכוחות המופעלים לאותות הפיקוד. בעת העברת קווי המגמה הליניאריים, עבור הגרפים בהם התלות היא במהירות סיבוב המנוע בריבוע קווי המגמה אולצו לעבור דרך הראשית מפני שברור כי אם המנוע אינו מסתובב לא יתקבל כוח דחף. לעומת זאת בעת העברת קווי המגמה הליניאריים עבור הגרפים בהם התלות היא באות הפיקוד קווי המגמה לא

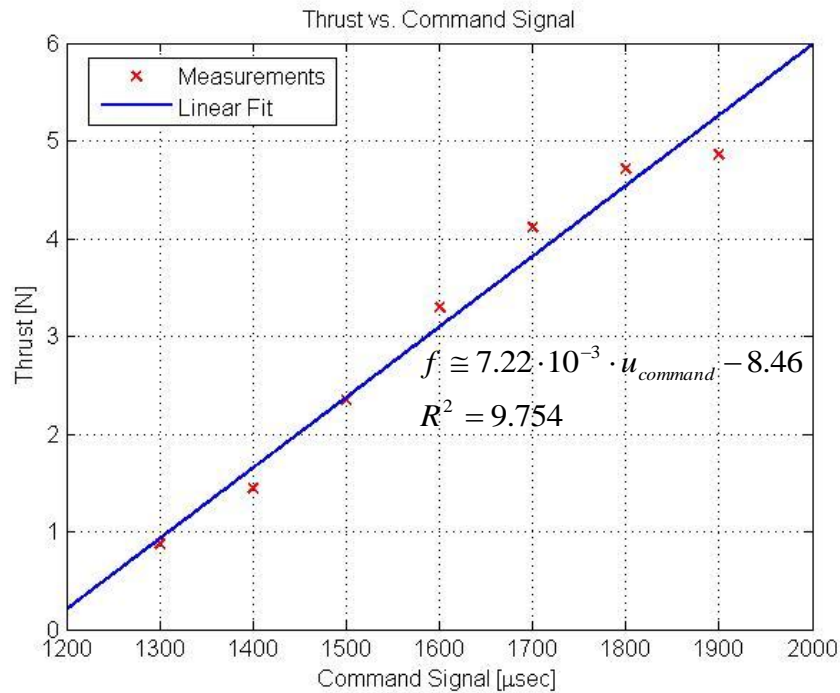
אולצו לעבור דרך הראשית, וזאת משום שיש צורך באות פיקוד גדול יחסית (≈ 1000) בכדי לגרום למנוע להתחיל להסתובב. להלן הגרפים אשר התקבלו:



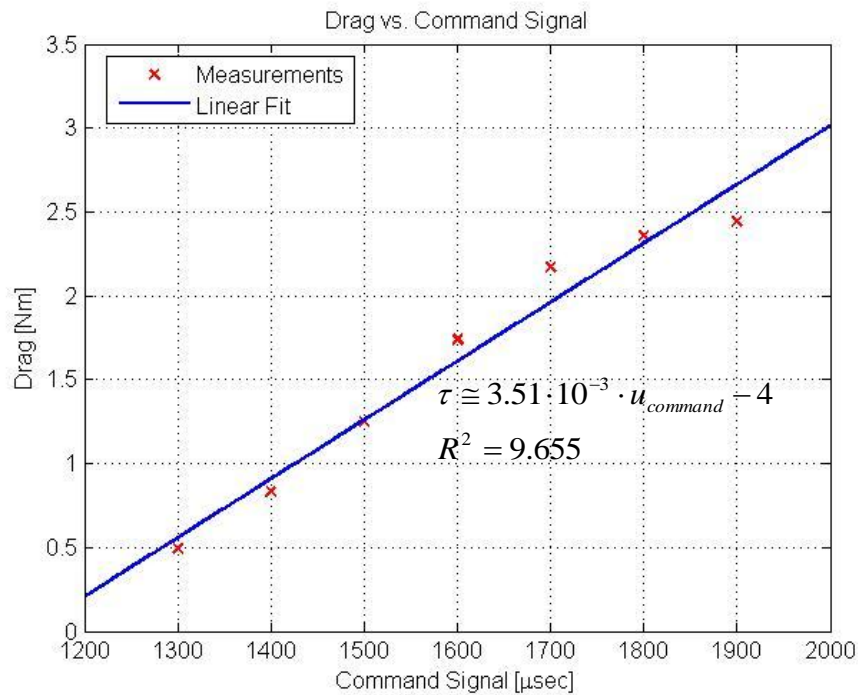
גרף א.5: כוח הדחף המופעל כנגד מהירות סיבוב המנוע בריבוע.



גרף א.6: כוח הגרר המופעל כנגד מהירות סיבוב המנוע בריבוע.



גרף א.7: כוח הדחף המופעל כנגד אות הפיקוד הנשלח למנוע.



גרף א.8: כוח הגרר המופעל כנגד אות הפיקוד הנשלח למנוע.

ניתן לראות מהגרפים בהם התלות היא במהירות סיבוב המנוע בריבוע כי קווי המגמה שהועברו הינם בעלי התאמה טובה מאוד ($R^2 > 9.9$), מכאן נוכל להסיק כי הקשר בין כוח הדחף וכוח העילוי לבין מהירות סיבוב המנוע בריבוע מציג בצורה טובה מאוד את המתרחש בפועל. לעומת זאת, עבור הגרפים בהם התלות היא באות הפיקוד הנשלח למנוע ניתן לראות כי ההתאמה פחות

טובה. אם נתבונן בגרפים נראה כי באותות פיקוד גבוהים ($u_{command} > 1700$) אנו מקבלים סטורציה. כמובן שתופעה זו הגיונית משום שיש גבול למהירות סיבוב המנוע שנוכל לקבל. לכן, בעת תכנון התנועה שתתבצע בניסוי נחסום את כוח הדחף וכוח הגרר הניתנים להפעלה בין אפס ועד לאילו המתקבלים בעת הפעלה של אות הפיקוד $u_{command} = 1700$.

לסיכום, כלל הקשרים שנמצאו עבור כוח העילוי ועבור כוח הגרר נתונים בטבלה הבאה:

כוח הדחף כתלות ב- ω^2	כוח הגרר כתלות ב- ω^2	כוח הדחף כתלות באות הפיקוד	כוח הגרר כתלות באות הפיקוד
$f = 3.89 \cdot 10^{-6} \cdot \omega^2$	$\tau = 2 \cdot 10^{-6} \cdot \omega^2$	$f = 7.22 \cdot 10^{-3} \cdot u_{command} - 8.46$	$\tau = 3.51 \cdot 10^{-3} \cdot u_{command} - 4$

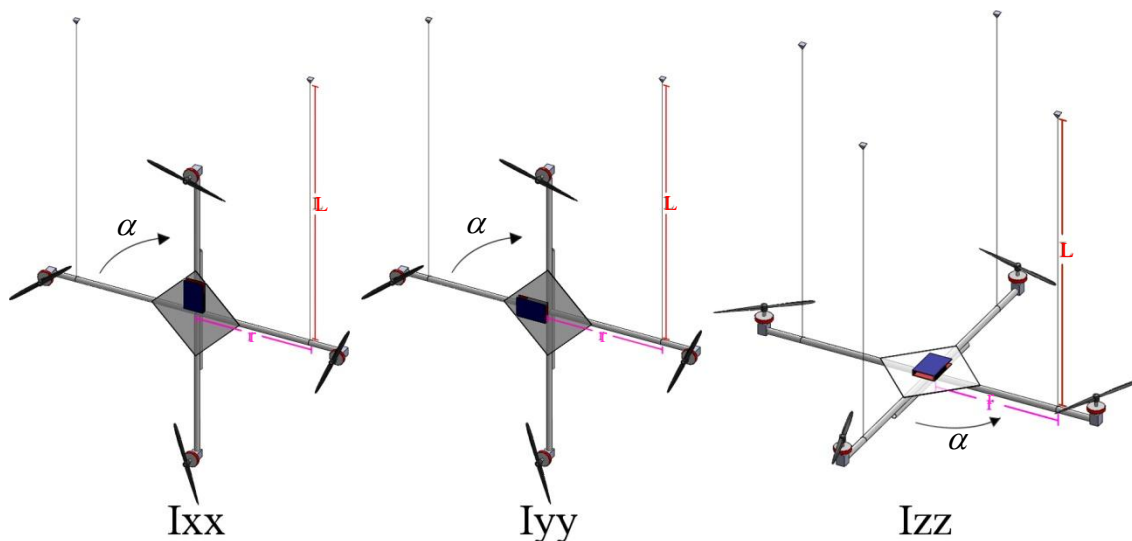
טבלה א.2: כלל הקשרים שהתקבלו בניסוי למציאת קבוע העילוי וקבוע הגרר.

א.2.2 מציאת מומנטי האינרציה של כלי הטיס

כדי לבצע את הסימולציות עבור תנועת המערכת באלגוריתם ה- RRT ובאלגוריתם ה- RRTT, ובכדי להשתמש בבקר שהוצג, יש למדוד את מומנטי האינרציה I_{xx} , I_{yy} , I_{zz} . נוכל למדוד פרמטרים אילו ע"פ השיטה המוצגת ב[12]. בשיטה זו מנדנד כלי הטיס סביב כל ציר בנפרד (ראה איור א.12) ונמדד תדר הנדנד. באמצעות ניתוח אנליטי פשוט עבור מטוטלת, תוך הנחת זוויות קטנות ושימוש במשפט הצירים המקבילים (שטיינר) נוכל לקבל את הקשר הבא:

$$I_{ii} = \frac{mgr^2}{\zeta_i^2 L} \quad (157)$$

כאשר I_{ii} הוא מומנט האינרציה ביחס למרכז הכובד סביב ציר i , r הוא המרחק בין נקודת הריתום לכלי למרכז הכובד (איור א.12), L הוא אורך חוט המטוטלת ו- ζ_i הוא תדר הנדנד סביב ציר i .



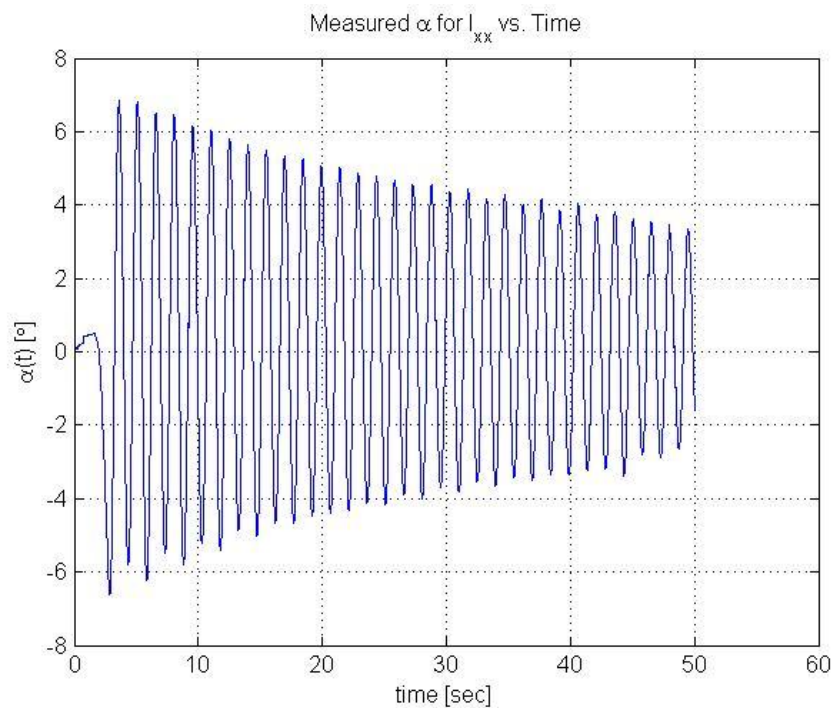
איור א.12: צורת חיבור המטוטלת לצורך מדידת מומנטי האינרציה של כלי הטיס.

במהלך ניסוי זה למציאת מומנטי האינרציה, כלי הטיס נתלה מן התקרה בשלושת צורות החיבור שהוצגו ונודנד בכיוון הנדרש:

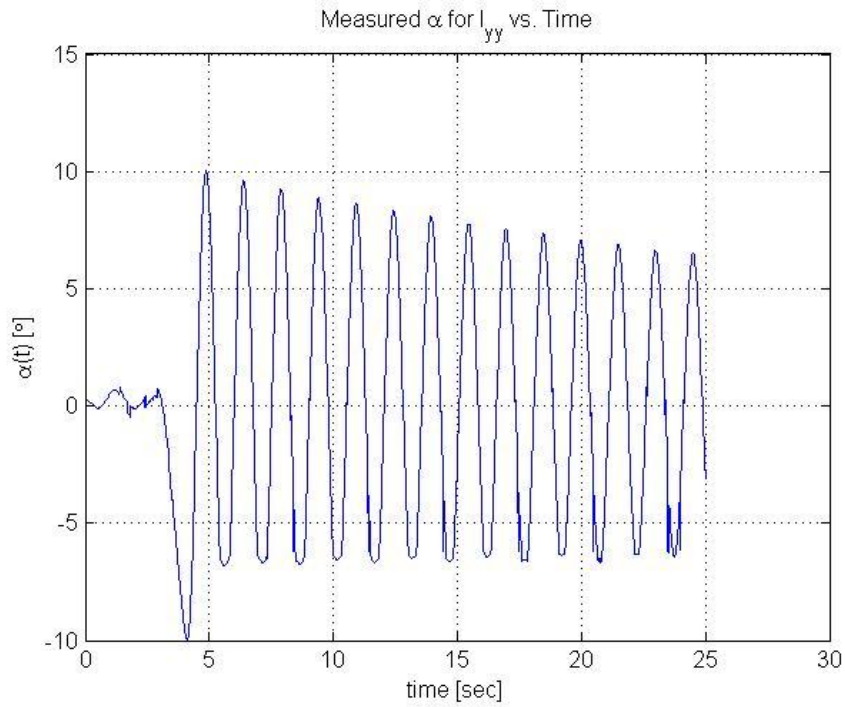


איור א.13: תמונה הממחישה את אופן תליית הכלי כמטוטלת.

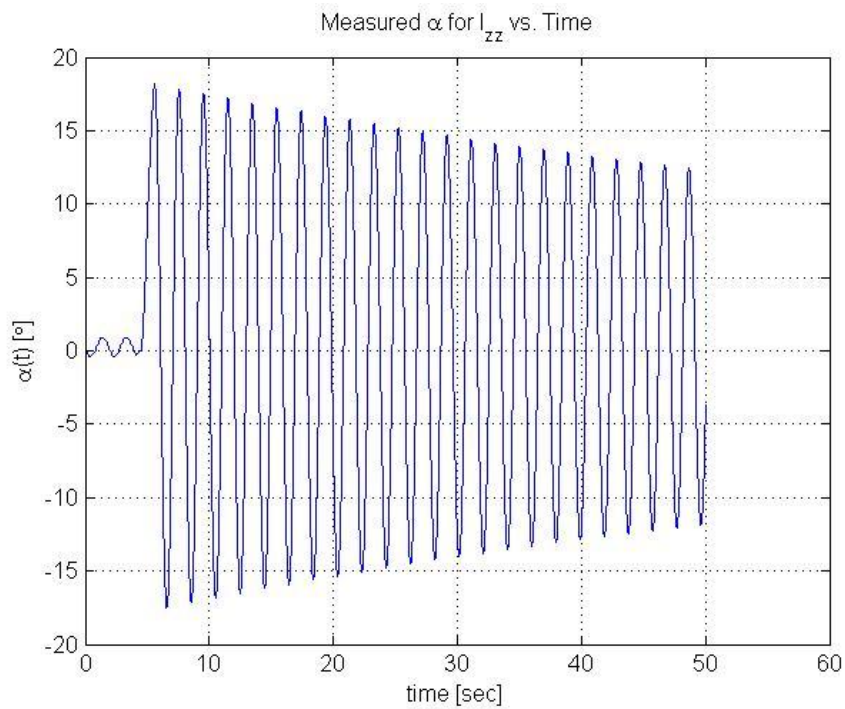
הפרמטרים r ו- L נמדדו באמצעות סרגל ומסת הכלי נמדדה באמצעות משקל דיגיטלי. בכדי למדוד את תדר הנדנוד של המטוטלת השתמשנו במערכת מצלמות הממוקמת במעבדה עליה נרחיב בהמשך. להלן התוצאות שהתקבלו עבור הזווית α (ראה איור א.12) כתלות בזמן:



גרף א.9: הזווית α כפונקציה של הזמן בכדי למצוא את I_{xx} .

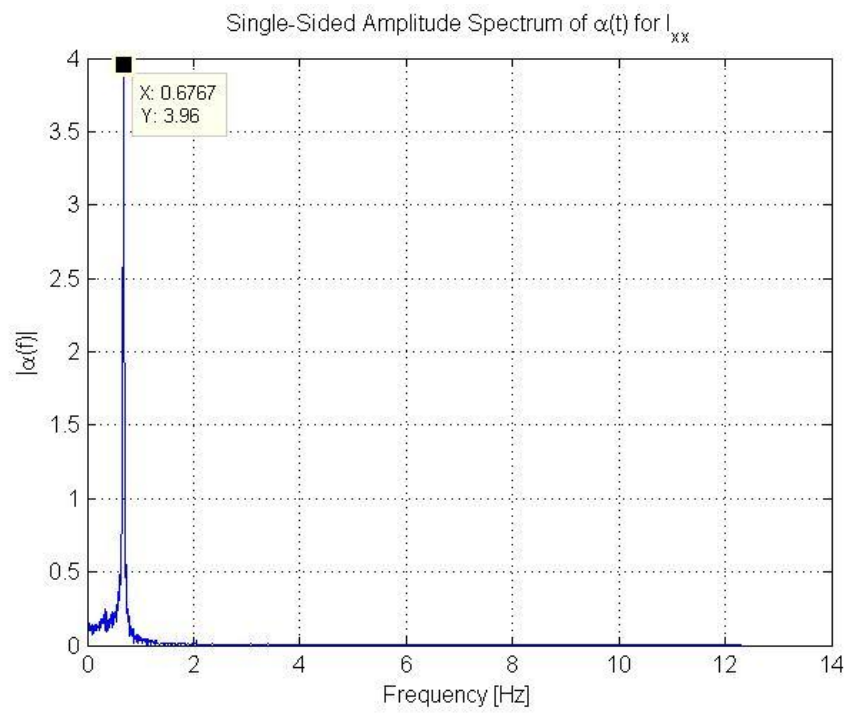


גרף א.10: הזווית α כפונקציה של הזמן בכדי למצוא את I_{yy} .

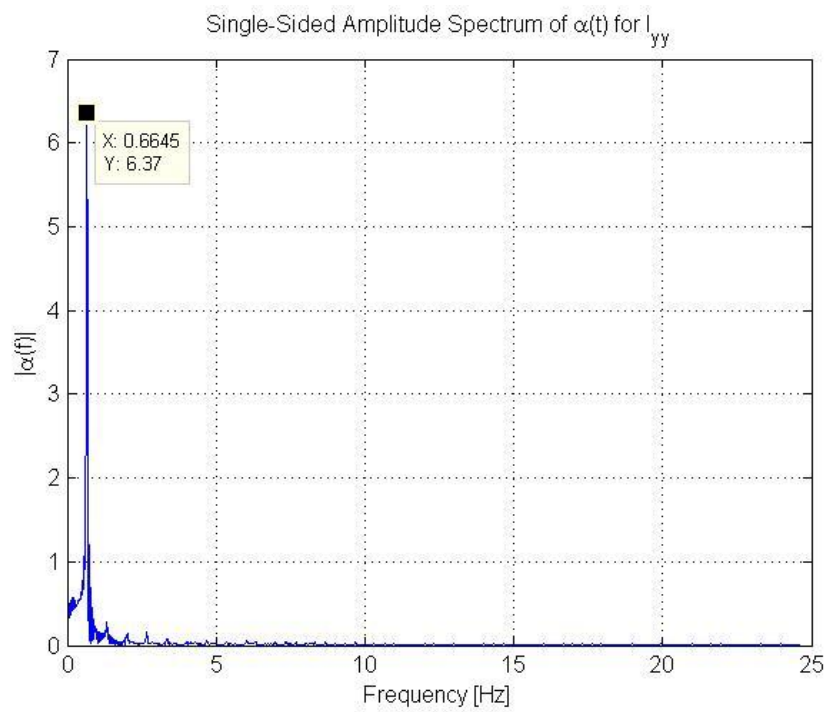


גרף א.11: הזווית α כפונקציה של הזמן בכדי למצוא את I_{zz} .

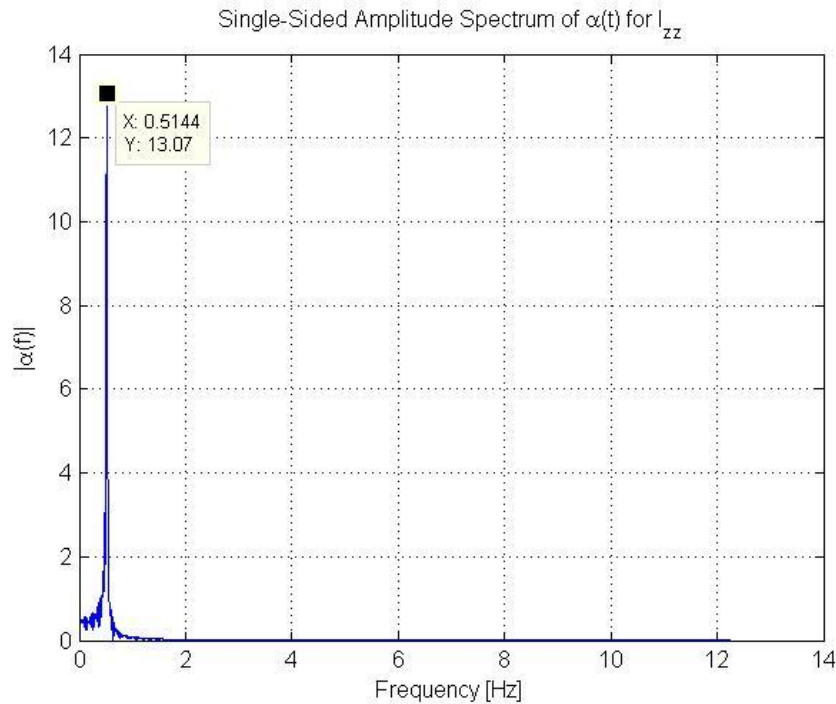
בכדי לקבל את תדר הנדנוד של הכלי סביב כל ציר נבצע התמרת פורייה לכל אחד מן הגרפים שהתקבלו. להלן הגרפים המתארים את תוצאות ההתמרות שבוצעו:



גרף א.12: תוצאות התמרת פורייה של הגרף עבור I_{xx} .



גרף א.13: תוצאות התמרת פורייה של הגרף עבור I_{yy} .



גרף א.14: תוצאות התמרת פורייה של הגרף עבור I_{zz} .

מהגרפים ניתן לראות כי תדרי הנדנוד שהתקבלו הם כדלהלן: $\zeta_z = 0.514 \text{ Hz}$, $\zeta_y = 0.664 \text{ Hz}$, $\zeta_z = 0.677 \text{ Hz}$. כלל הפרמטרים שנמדדו בכדי להשתמש בנוסחא (157) מוצגים בטבלה הבאה:

l [m]	r [m]	m [kg]	ζ_z $\left[\frac{\text{rad}}{\text{sec}}\right]$	ζ_y $\left[\frac{\text{rad}}{\text{sec}}\right]$	ζ_x $\left[\frac{\text{rad}}{\text{sec}}\right]$
1.805	0.25	0.8	3.23	4.17	4.25

טבלה א.3: כלל הפרמטרים שנמדדו בכדי לחשב את מומנטי האינרציה של הכלי.

לאחר חישוב לפי נוסחא (157) ועל סמך המדידות שבוצעו, התקבלו מומנטי האינרציה הבאים עבור כלי הטיס:

I_{zz} $[kg \cdot m^2]$	I_{yy} $[kg \cdot m^2]$	I_{xx} $[kg \cdot m^2]$
0.021820	0.013075	0.012578

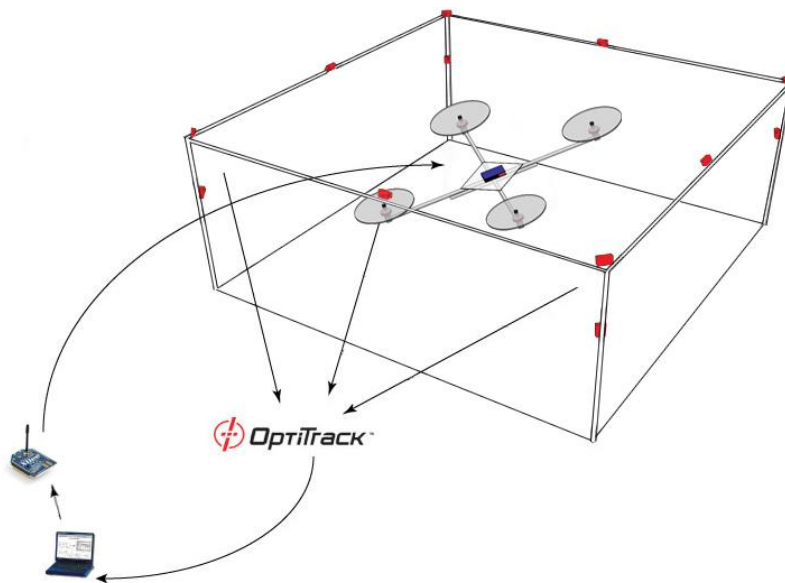
טבלה א.4: מומנטי האינרציה שהתקבלו עבור כלי הטיס.

ניתן לראות מן התוצאות שהתקבלו כי מומנטי האינרציה I_{xx} ו- I_{yy} שהתקבלו הינם קרובים בערכם. תוצאה הגיונית משום שמלבד האופן בו מונח המיקרו-בקר על כלי הטיס (איור א.12) ומלבד הצורה בה מחוברת הסוללה, גוף הכלי זהה ביחד לשני הצירים x ו- y . כמו כן ניתן לראות כי מומנט האינרציה סביב ציר z הינו הגדול ביותר (כמעט פי שניים משני מומנטי האינרציה האחרים). תוצאה זו גם כן הגיונית משום שמרבית המסה מונחת על המישור הניצב לציר z .

3.א. מערכת הניסוי

מערכת הניסוי כוללת את כלי הטיס שהוצג, מחשב PC נייח ומערכת מצלמות. במעבדה בה נערך המחקר קיימת מערכת מצלמות מתוצרת חברת OptiTrack הכוללת שתיים-עשרה מצלמות, אשר מפוזרות במעבדה בצורה שתביא לתפיסת נפח מירבית. שימוש במצלמות אילו, יחד עם תוכנה ייעודית, מאפשר עקיבה אחר סמנים אופטיים (למעשה מחזירי אור) המודבקים לגוף כלשהו אחריו רוצים לעקוב. באמצעות מערכת זו ניתן למעשה לקבל בזמן אמת את מיקום כלי הטיס בדיוק גבוה ממילימטר, כמו גם את האוריינטציה של כלי הטיס. נתונים אילו הכרחיים לשימוש בבקר שהוצג בכדי לממש את תכנוני התנועה שבוצעו.

במהלך הניסוי, כלי הטיס הוטס באמצעות שלט אל נקודת ההתחלה ובאמצעות שינוי מצב בשלט, עבר הכלי למצב טיסה אוטונומי. על המחשב נמצאת תוכנה המבצעת את הבקרה בהתאם לבקר שהוצג, כאשר את המיקום והאוריינטציה הנוכחיים התוכנה מקבלת מן המצלמות, ואת המיקום והאוריינטציה (כמו גם המהירויות) הרצויות התוכנה מקבלת מתכנון התנועה שהתבצע מראש. לאחר חישוב הכניסות הרצויות, התוכנה הנ"ל שולחת באמצעות Xbee את ארבעת הכניסות הרצויות אל כלי הטיס אשר מפעיל את הכניסות הללו על-סמך ההמרה מכוח לאות פיקוד. ניתן לראות באיור א.14 את מבנה מערכת הניסוי:



איור א.14: מבנה מערכת הניסוי.

4.א. תוצאות הניסויים

במהלך הניסוי הראשון בוצע ניסיון לבצע את התנועה שתוכננה בסימולציה הראשונה עבור אלגוריתם ה-RRT (ראה איור 6.13). כעבור כשנייה לאחר מעבר למצב אוטונומי, כלי הטיס החל להתנהג בצורה משונה והתקבלה החלטה לבצע נחיתת חירום (אין רשתות הגנה במעבדה). בעת הנחיתה המיקרו-בקר אשר על כלי הטיס קיבל מכה חזקה והפסיק לעבוד כנדרש - צורת השליחה של הפקודות למנועים נפגמה ואינה עובדת. לאחר אירוע זה הוחלט שלא לבצע ניסויים נוספים.

הסיבות לכישלון הניסוי הן ככל הנראה שגיאות המדידה הקיימות במהלך הניסויים למציאת מומנטי האינרציה של כלי הטיס ובמהלך הניסויים למציאת קבוע העילוי וקבוע הגרר. כמו כן, כאשר הבקר מורה לכלי הטיס להפעיל כוח מסויים במנוע מס' 1 לדוגמא, על כלי הטיס לשנות את מהירות סיבוב המנוע עד להגעה למהירות הרצויה - כלומר במהלך הניסוי, בניגוד לסימולציה הממוחשבת, עובר פרק זמן מסויים בו המנוע מפעיל כוח אשר שונה מן הכוח הרצוי. כלל השגיאות הללו מצטברות, ובכל סימולציה להתקדמות בזמן במהלך תכנון התנועה ישנה שגיאה נוספת. השגיאות של מומנטי האינרציה זניחות, כך שבקרה על המנועים, תוך התחשבות בפונקציית התמסורת של המנועים עשויה לאפשר ניסוי מוצלח.

בנוסף, ככל שזמן הטיסה ארוך יותר כך ההספק שמסוגלת הסוללה לנפק יורד. עובדה זו גורמת לכך ששליחה של אות פיקוד מסויים בתחילת הטיסה תיתן כוח שונה משליחה של אות פיקוד זהה לאחר שכלי הטיס נמצא באוויר פרק זמן מסויים. הגבלת הכוחות המקסימליים במהלך הסימולציה יכולה לפתור בעיה זו. אם נתבונן בגרפים א.7 ו - א.8 נראה כי עבור אותו פיקוד גדולים יחסית ישנו מצב של סטורציה, ולכן ע"י הגבלה על גובה אות הפיקוד (ועל גודל הכוחות) נקבל את אותו הכוח גם לאחר טיסה וירידת הספק הסוללה עבור אות פיקוד זהה.

Abstract

In this report we showed the modeling of a four thrusters aircraft (Quadrotor) dynamics while developing fully its equations of motion. In addition, we presented a non-linear controller that is based on differential geometry methods, which enables tracking a pre-defined path. Then, we showed a classic method of path planning that is based on the Generalized Voronoi Graph (GVG) roadmap, which is constructed in the configuration space, while using Dijkstra's algorithm. In addition, we presented the RRT algorithm for motion planning in the state space, and we introduced an improved algorithm for motion planning in the state space - the RRTT algorithm. Finally, we showed simulation results, and explained about the experiment system that was established.

The quadrotor is an unmanned aerial vehicle (UAV) that is based on a cross with four motors on it, which turn rotors. First we modeled the dynamics of the aircraft and developed its complete motion equations, while taking into consideration the gyroscopic effects, the rotation angles of each rotor and the model of the rotors. We derived these equations of motion using the Euler-Lagrange method. Then, we assumed a number of assumptions and found a simpler model.

The quadrotor has four input forces and six degrees of freedom. In other words, the quadrotor is an under-actuated system. Under-actuated systems are harder to control because there is a coupling between the different degrees of freedom. In this report we presented a non-linear controller, that is based on differential geometry methods, and consists of three different flight modes. Each flight mode achieves almost global exponential tracking of four different degrees of freedom. We proved that the controller is stable, and performed simulations.

After presenting the control law, we showed and explained a classic path planning method for a known work space. We presented the Generalized Voronoi Graph - a roadmap that is constructed in the configuration space (usually). In addition, we showed and explained Dijkstra's algorithm in order to find the shortest path from the starting point to the goal point on the GVG. We performed simulations, proved that the GVG is in fact a roadmap, proved the correctness of Dijkstra's algorithm while

computing its complexity, and showed the disadvantages of path planning as appose to motion planning (trajectoy planning).

In addition, we presented the RRT algorithm for motion planning in the state space, while thoroughly explaining how it works, proving that it is probabilistically complete and trying to estimate the number of iterations needed in order to compute a motion solution. After understanding the advantages and disadvantages of the RRT algorihm, we introduced a new and improved algorithm - the RRTT algorithm. We proved that it is also probabilistically complete, and showed that for a standard work space the needed number of iterations for computing a trajectory will probably be less than the number needed by the RRT. In addition, we of course performed simulation of using these two algorithms.

Finally, we presented the building process of the quadrotor. Then we explained the different experiments that were held in order to compute the aircraft's parameters. Finally, we presented the entire experiment system that was established.



**BEN-GURION UNIVERSITY OF THE NEGEV
FACULTY OF ENGINEERING SCIENCES
DEPARTMENT OF MECHANICAL ENGINEERING**

***CONTROL AND MOTION PLANNING
OF A QUADROTOR UAV***

Thesis Submitted in Partial Fulfillment of the Requirements
for the M.Sc. Degree

BY: Shlomo Shemesh

SUPERVISOR: Dr. Amir Shapiro

Author:

Supervisor:

Chairman of Graduate Studies Committee:

September 2012



**BEN-GURION UNIVERSITY OF THE NEGEV
FACULTY OF ENGINEERING SCIENCES
DEPARTMENT OF MECHANICAL ENGINEERING**

***CONTROL AND MOTION PLANNING
OF A QUADROTOR UAV***

Thesis Submitted in Partial Fulfillment of the Requirements
for the M.Sc. Degree

BY: Shlomo Shemesh

September 2012